NASA TECHNICAL HANDBOOK	NASA-HDBK-0008
National Aeronautics and Space Administration Washington, DC 20546-0001	Approved: 12-17-2012

NASA PRODUCT DATA AND LIFE-CYCLE MANAGEMENT (PDLM) HANDBOOK

MEASUREMENT SYSTEM IDENTIFICATION:

NOT MEASUREMENT SENSITIVE

DOCUMENT HISTORY LOG

Status	Document Revision	Approval Date	Description
Baseline		12-17-2012	Initial Release

FOREWORD

This Handbook is published by the National Aeronautics and Space Administration (NASA) as a guidance document to provide engineering information; lessons learned; possible options to address technical issues; classification of similar items, materials, or processes; interpretative direction and techniques; and any other type of guidance information that may help the Government or its contractors in the design, construction, selection, management, support, or operation of systems, products, processes, or services.

This Handbook is approved for use by NASA Headquarters and NASA Centers, including Component Facilities and Technical and Service Support Centers.

It provides information on product data and life-cycle management (PDLM) and general guidance to adapt the methods needed to implement the requirements in NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects. This Handbook also addresses elements to consider when developing the PDLM Plan so that enablers and users of the product are better equipped to recognize and avoid pitfalls that might otherwise be experienced.

Requests for information, corrections, or additions to this Handbook should be submitted via "Feedback" in the NASA Standards and Technical Assistance Resource Tool at https://standards.nasa.gov.

Original Signed By:	**12-17-2012**
Michael G. Ryschkewitsch NASA Chief Engineer	Approval Date

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

APPENDICES

LIST OF FIGURES

LIST OF TABLES

NASA PRODUCT DATA AND LIFE-CYCLE MANAGEMENT (PDLM) HANDBOOK

1. SCOPE

1.1 Purpose

The purpose of this Handbook is to provide information on product data and life-cycle management (PDLM) and general guidance to adapt the methods needed to implement the requirements in NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects. This Handbook also addresses elements to consider when developing the PDLM Plan (refer to the template in NPR 7120.9, Appendix D) so that enablers and users of the product are better equipped to recognize lessons learned and avoid pitfalls that might otherwise be experienced.

The creation, management, and usage of product-related data across a cradle-to-grave life cycle are daily events at NASA. The integration and sharing of electronic product data among Centers, across programs/projects, and with primes and subcontractors have become mission critical. Multi-disciplinary teams (such as systems engineering, product engineering, manufacturing, purchasing, etc.), as well as remote participants (e.g., local or globally disbursed suppliers, subcontractors, etc.), need quick access to (1) the product data on which they are working, and (2) the associated information that better defines product performance, functionality, form, and fit. The amount, type, and fidelity of the data generated and requiring storage increase over the program/project life cycle. The scale and complexity of the storage and retrieval system have to respond to meet the challenges as shown in figure 1, Data over the Program/Project Life Cycle (Sample Only—Not All Inclusive). The linkage between first managing product data at the Center (or authoring stage) during the key activities of design engineering and manufacturing before extending its communication and sharing across the longer life cycle has led to the adoption of the phrase "product data and life-cycle management," or PDLM, as shorthand for the more cumbersome product data management/product life-cycle management (PDM/PLM) label. (Definitions of PDM and PLM in Section 3.2 explain these terms individually.) PDLM includes configuration management (CM), requirements management, risk management, and other artifacts developed within the product life cycle; therefore interoperability from beginning of life to end of life is a staple and a requirement of PDLM.

Product data management (PDM) provides key capabilities that underlie product life-cycle management (PLM) and is widely considered a precursor requirement to effective PDLM. PDLM does not negate product data management. A process for management of technical data is required by NPR 7123.1, NASA Systems Engineering Processes and Requirements, for ensuring that the data required are captured and stored, data integrity is maintained, and data are disseminated as required. PDLM augments PDM and addresses the acquisition and storage of all program/project data—not just technical data used in the life cycle of a system.

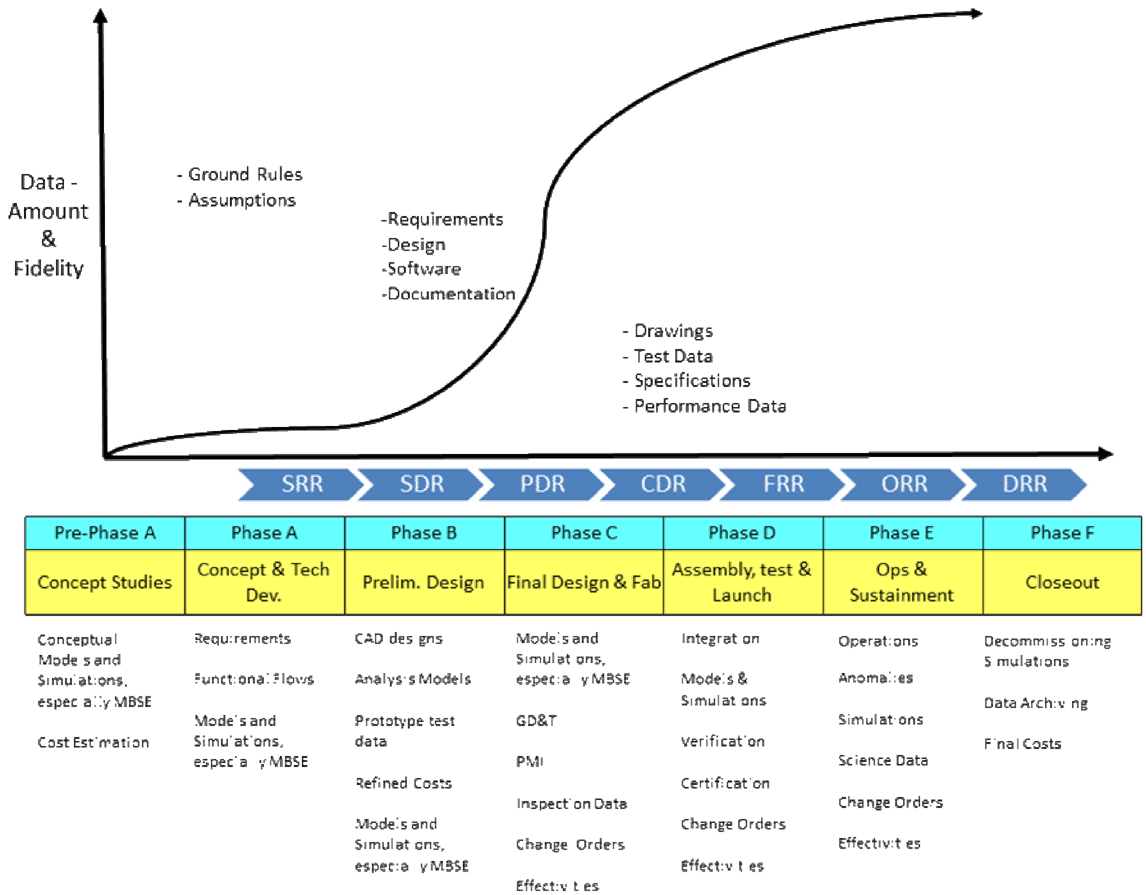


Figure 1—Data over the Program/Project Life Cycle (Sample Only—Not All Inclusive)

PDLM consists of disciplined, collaborative processes and systems that plan for, acquire, and control product definition data (PDD) and associated product-related data, including engineering, design, test, procurement, manufacturing, operational, and logistics information throughout the product and data life cycles. PDLM is the set of processes and associated information used to manage the entire life cycle of product data from its conception, through design, test, and manufacturing, to service and disposal. To do so requires managing the creation and changes to product definition, product configurations, affiliated engineering data, data on the performance of the product components in mission environments, and product software and hardware.

As indicated in figure 2, PDLM Framework, PDLM is the framework that integrates data, processes (elements), tools, and business systems to provide users with a product information backbone for NASA programs/projects. A life-cycle-oriented approach to PDLM is intended to reduce or eliminate redundant development activities, increase collaborative design and analysis, and reduce time to complete informed decision making throughout the program/project life cycle. PDLM can be implemented using the traditional document-centric approach or with a hybrid approach where both modeling and documents are combined to meet the program's/project's PDLM needs. To achieve the highest level of collaboration, interoperability, efficiency, and data availability, the implementation of a fully digital PDLM environment, referred to as a PDLM Collaborative Environment (PCE), as defined in this Handbook, is necessary.

PDLM requires that information technology (IT) systems across NASA be made interoperable or integrated to the extent needed to provide a secure, readily accessible environment to enable required collaborative PDLM capabilities. PDLM is intended to increase the probability of mission success by increasing the availability, effectiveness, and efficiency of data interchange and integration across disparate systems and the availability of the right data for the right people at the right time, thereby reducing risk.

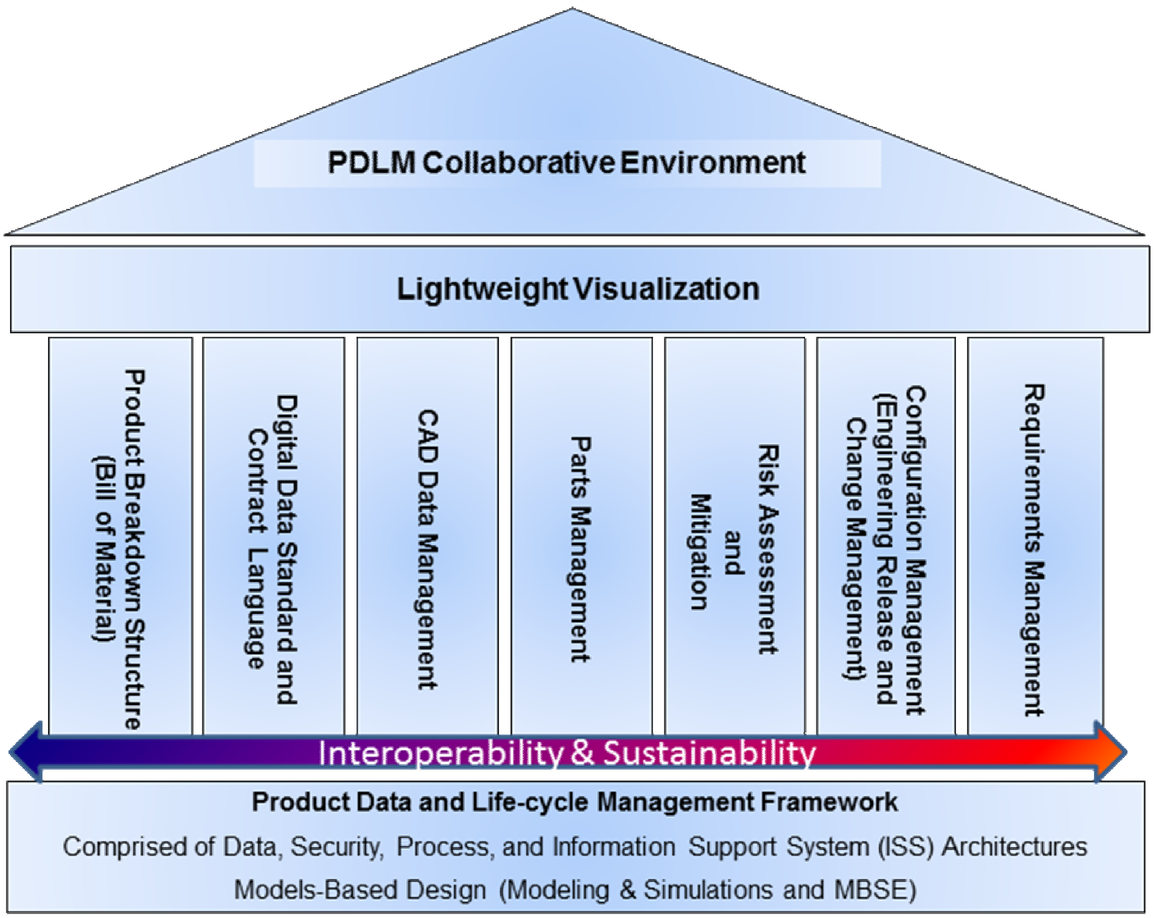


Figure 2—PDLM Framework

1.2 Applicability

This Handbook is applicable to the following:

a. Current and future NASA space flight single-project and tightly coupled programs and their projects subject to NPR 7120.9.

b. It is also recommended that this Handbook be considered for guidance in all other NASA programs/projects.

This Handbook is approved for use by NASA Headquarters and NASA Centers, including Component Facilities and Technical and Service Support Centers. This Handbook may also apply to the Jet Propulsion Laboratory or to other contractors, grant recipients, or parties to agreements only to the extent specified or referenced in their contracts, grants, or agreements.

This Handbook, or portions thereof, may be referenced in contract, program/project, and other Agency documents for guidance. When this Handbook contains procedural or process requirements, they may be cited in contract, program/project, and other Agency documents.

2. APPLICABLE DOCUMENTS

2.1 General

The documents listed in this section are applicable to the guidance in this Handbook.

2.1.1 The latest issuances of cited documents shall apply unless specific versions are designated.

2.1.2 Non-use of specific versions as designated shall be approved by the responsible Technical Authority.

The applicable documents are accessible via the NASA Standards and Technical Assistance Resource Tool at https://standards.nasa.gov or may be obtained directly from the Standards Developing Organizations or other document distributors.

2.2 Government Documents

National Aeronautics and Space Administration

NPR 2800.1	Managing Information Technology
NPR 2810.1	Security of Information Technology
NPR 7120.9	NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects
NPR 7123.1	NASA Systems Engineering Processes and Requirements
NPR 7150.2	NASA Software Engineering Requirements
NASA-STD-0007 (including the interim version)	NASA Computer-Aided Design Interoperability
NASA-STD-7009	Standard for Models and Simulations

2.3 Non-Government Documents

None.

2.4 Order of Precedence

This Handbook provides guidance on PDLM to adapt the methods needed to implement the requirements in NPR 7120.9 and addresses elements to consider when developing the PDLM Plan but does not supersede nor waive established Agency requirements/guidance found in other documentation.

3. ACRONYMS, ABBREVIATIONS, AND DEFINITIONS

3.1 Acronyms and Abbreviations

2D	two dimensional
3D	three dimensional
AP	application protocol
API	application programming interface
BOM	bill of material
CAD	computer-aided design
CAE	computer-aided engineering
CAM	computer-aided manufacturing
CDR	Critical Design Review
CGM	computer graphics metafile
CIO	Chief Information Officer
CM	configuration management
COLLADA	COLLAborative Design Activity
COTS	commercial off-the-shelf
CSV	comma-separated value
DDT&E	design, development, test, and evaluation
DMU	digital mock-up
DoD	Department of Defense
DRD	Data Requirements Description
DRR	Disposal Readiness Review
DXF	drawing exchange format
eCAD	electrical computer-aided design
ePaper	electronic paper
ECM	Enterprise Content Management
EEE	electronic, electrical, and electromechanical
ETL	Extraction-Transformation-Loading (also EXTL)
EXTL	Extraction-Transformation-Loading (also ETL)
FIPS	Federal Information Processing Standard
FRR	Flight Readiness Review
GD&T	Geometric Dimensioning and Tolerancing
GN&C	Guidance, Navigation, and Control

GIF	Graphics Interchange Format
GOTS	Government off-the-shelf
GSE	ground support equipment
HDBK	handbook
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IGES	Initial Graphics Exchange Specification
INCOSE	International Council on Systems Engineering
ISO	International Organization for Standardization
ISSA	Information Support System Architecture
IT	information technology
JPEG/JPG	Joint Photographic Experts Group
KDP	key decision point
LOTAR	Long Term Archiving and Retrieval
LOX	liquid oxygen
M&S	models and simulation
MBSE	models-based systems engineering
mCAD	mechanical computer-aided design
MOF	Meta Object Facility
MOTS	modified off-the-shelf
MSFC	Marshall Space Flight Center
NASA	National Aeronautics and Space Administration
NIST	National Institute of Standards and Technology
NPD	NASA Policy Directive
NPR	NASA Procedural Requirements
OMG	Object Management Group
ORR	Operational Readiness Review
PCE	PDLM Collaborative Environment
PDD	product definition data
PDF	portable document format
PDLM	product data and life-cycle management
PDM	product data management
PDM/PLM	product data management/product life-cycle management
PDR	Preliminary Design Review
PLM	product life-cycle management
PMI	product and manufacturing information
PNG	portable network graphics
REST	representative state transfer
RFP	Request for Proposal
S&MA	Safety and Mission Assurance
SBU	sensitive but unclassified
SDR	System Definition Review
SGML	standard generalized markup language
SOAP	simple object access protocol
SQL	structured query language
SRR	System Requirements Review

STD	Standard
STEP	Standard for the Exchange of Product
SysML™	Systems Modeling Language™
TBD	to be determined
UID	unique identification
UML	unified modeling language
URI	uniform research identifier
W3C	World Wide Web Consortium
X3D	extensible 3D
XMI	XML metadata interchange
XML	Extensible Markup Language
XSD	XML Schema

3.2 Definitions

Authoritative Data: Data that has been designated as valid for specific official programs/projects. The designated data is controlled by processes. (Source: NPD 7120.4, NASA Engineering and Program/Project Management Policy)

Authoritative Source: An application or repository identified as the official source for specific authoritative data. (Source: Adapted from NPD 7120.4, NASA Engineering and Program/Project Management Policy definition of Authoritative Data)

Bill of Material (BOM): A listing for a semi-finished or finished product (e.g., end item) containing component parts and materials making up a single instance of a product with a name, reference, or part number, quantity, and unit of measure for each component. Note: A BOM is a representation of product information for a particular purpose such as engineering, manufacturing, procurement, or sustainment. An indentured, hierarchical BOM is a form of product breakdown structure because it captures product component relationships as well as identity and quantity. (Adapted from Department of Defense (DoD) and industry sources)

Computer-Aided Design (CAD): Process of creating engineering designs defined by electronically produced multi-dimensional geometry using special software systems, the tools used by engineers to create geometric-based design definitions represented in a variety of formats, including two-dimensional (2D) drawings, three-dimensional (3D) solid models, envelopes, wireframes, and kinematics and time-based models.

Configuration Management (CM): A management discipline applied over a product's life cycle to provide visibility into and to control changes to performance, functionality, and physical characteristics. (Source: NPR 7120.5, NASA Space Flight Program and Project Management Requirements) It is also a process that establishes and maintains consistency of a product's attributes with the requirements and product configuration information throughout the product's life cycle. (Source: NASA-STD-0005, NASA Configuration Management (CM) Standard)

Data: A representation of facts, concepts, or instructions in a manner suitable for communication, interpretation, or processing by humans or by automatic means. (Source: ISO/IEC 24765:2009, Systems and software engineering vocabulary)

Data Architecture: Provides an understanding of what information is needed to effectively execute the Enterprise's business processes and provides a framework for effectively managing the Enterprise's information environment. Note: Data Architecture links information behavior (i.e., accessing, using, and sharing data), information management processes, and information support staff to other aspects of the Enterprise. (Source: NPR 2830.1, NASA Enterprise Architecture Procedures)

Data Life Cycle: The series of states that a data object can take from its creation to its retirement or destruction; generally, these states represent maturity levels or indicate suitability for, or restrictions on, use. (Source: NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects)

Data Management: The process used to:

a. Provide the basis for identifying and controlling data requirements.

b. Responsively and economically acquire, access, and distribute data needed to develop, manage, operate, and support system products over their product-line life.

c. Manage and dispose data as records.

d. Analyze data use.

e. If any of the technical effort is performed by an external contractor, obtain technical data feedback for managing the contracted technical effort; and

f. Assess the collection of appropriate technical data and information.

g. Effectively manage authoritative data that defines, describes, analyzes, and characterizes a product life cycle.

h. Ensure consistent, repeatable use of effective PDLM processes, best practices, interoperability approaches, methodologies, and traceability. (Source: NPR 7123.1, NASA Systems Engineering Processes and Requirements)

Data Model: Identifies the data, their attributes, and relationships or associations with other data. (Source: Department of Defense Directive 8320.03, Unique Identification (UID) Standards for a Net-Centric Department of Defense, March 2007)

Delivery: Applies to the hand-off, which may be physical or virtual in the case of data, at initial delivery of an item during development, for collaboration, in support of reviews, at the time of acceptance, and for subsequent modifications, maintenance, refurbishment, or any other activity that produces new data. (Source: NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects)

End Item: A product to be delivered under contract as an intact unit or to be assembled, completed, and made ready for use as a unit. (Source: NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects)

Engineering Release: An action whereby engineering documentation or an item is officially made available for the intended use. (Source: Modified from NASA-STD-0005, NASA Configuration Management (CM) Standard)

Firmware. The combination of a hardware device and computer instructions and data that reside as read-only software on that device. Notes: (1) This term is sometimes used to refer only to the hardware device or only to the computer instructions or data, but these meanings are deprecated. (2) The confusion surrounding this term has led some to suggest that it be avoided altogether. (Source: IEEE-STD-610.12-1990, IEEE Standard Glossary of Software Engineering Terminology)

Governance Model: The framework—principles and structures—through which the Agency manages the mission, roles, and responsibilities. (Source: NPD 1000.0, NASA Governance and Strategic Management Handbook)

Metadata: Data about data, including information describing aspects of actual data items, such as name, type, format, content, and other descriptive information. (Source: NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects)

Model: A description or representation of a system, entity, phenomena, or process. (Source: NASA-STD-7009, Standard for Models and Simulations)

Product: A part of a system consisting of end products that perform operational functions and enabling products that perform life-cycle services related to the end product or a result of the technical efforts in the form of a work product (e.g., plan, baseline, or test result). (Source: NPR 7123.1, NASA Systems Engineering Processes and Requirements) Note: Products include, but are not limited to, spacecraft, launch vehicles, instruments, payloads, software, launch platforms, and other elements that are necessary components to deliver a completed vehicle.

Product Breakdown Structure: A hierarchical view of the relationship of products and component products. (Source: NPD 7120.4, NASA Engineering and Program/Project Management Policy)

Product Data and Life-Cycle Management (PDLM) System: A combination of the information technology applications, users, and processes that implement the management of product data across the product life cycle. (Source: NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM) for Flight Programs and Projects)

<u>Product Data Management (PDM)</u>: The framework that enables organizations to manage and control engineering and technical information, specifically data surrounding the product's design, definition, and related engineering, manufacturing, and logistics processes and is a key element of PLM. Note: From the product perspective, PDM organizes data required for design evolution, tracks versions and configurations of evolving design concepts, and manages archived data and other product-specific information. PDM tools provide access to product structures and other engineering data such as requirements, as-built, and safety and mission assurance data. From the process perspective, PDM systems offer the capability to orchestrate controlled procedural events, such as design reviews, approvals, product releases, and configuration audits. (Source: NPD 7120.4, NASA Engineering and Program/Project Management Policy)

<u>Product Definition Data (PDD)</u>: The data objects and associated elements required to completely define a product. Note: In normal usage, PDD refers to the authoritative design engineering design definition, but it can include data associated with production, operations, maintenance, and disposal. (Source: Adapted from ISO 16792, Technical product documentation—Digital product definition data practices)

<u>Product Life Cycle</u>: A series of states that generally defines the maturity level of a product and correlates with specific uses or users. Commonly, each state is represented by an agreed-to collection of information that identifies and establishes the attributes of a product at a point in time and that serves as the basis for defining change. Note: A product's life cycle begins with a concept and ends with disposal. (Source: NPR 7120.9, NASA Product Data and Life-Cycle Management (PDLM)) for Flight Programs and Projects)

<u>Product Life-Cycle Management (PLM)</u>: A strategic business approach that applies a consistent set of business solutions in support of the collaborative creation, management, dissemination, and use of product definition data/information across the extended enterprise from concept to end of life. Note: PLM integrates people/organizations, processes, and information. In product-dominated endeavors, PLM serves as the information backbone that extends outside the enterprise. PLM implementations may be composed of multiple elements, including foundation technologies and standards (e.g., Extensible Markup Language (XML)), visualization, collaboration, and enterprise application integration), information authoring tools (e.g., mechanical computer-aided design, electrical computer-aided design, and technical publishing), core functions (e.g., data vaults, document and content management, workflow and program/project management), functional applications (e.g., CM), and business solutions built on the other elements. (Source: NPD 7120.4, NASA Engineering and Program/Project Management Policy)

<u>Simulation</u>: The imitation of the characteristics of a system, entity, phenomena, or process using a computational model. (Source: NASA-STD-7009, Standard for Models and Simulations)

<u>Software</u>: Computer programs, procedures, rules, and associated documentation and data pertaining to the development and operation of a computer system. Software also includes but is not limited to commercial off-the-shelf (COTS) software, Government off-the-shelf (GOTS) software, modified off-the-shelf (MOTS) software, embedded software, reuse, heritage, legacy,

auto-generated code, firmware, and open source software components. (Source: NPD 7120.4, NASA Engineering and Program/Project Management Policy)

Virtual Database: A container for components used to integrate data from multiple data sources so that they can be accessed in an integrated manner through a single, uniform application programming interface (API). It contains models, which define the structural characteristics of data sources, views, and Web services.

4. PDLM STRATEGY

The underlying premise of PDLM is that NASA gets the "right data to the right people at the right time," so sustainable PDLM and collaboration for engineering design data for representative types of projects are key efforts.

The PDLM environment and architectural design are based on a series of key considerations:

The proper use of embedded capabilities that match approved processes and coverage delivered by technology in place at NASA (leverage the existing infrastructure where it supports the model).

a. Clear definition of the components that define the nature and extent of PLM is relevant for NASA.

b. PDLM cannot be considered one tool but a configuration of capabilities selected, integrated, and managed to fit the program's, project's, or organization's needs. PDLM tools have out-of-the-box capabilities but are not out-of-the-box systems. They are complex combinations of applications with their own extensive "configurable" capabilities and interfaces and integrations with and among other applications.

c. PDLM builds on the core capability of an organization to create and manage design data at the engineering-to-manufacturing interface. At NASA, this interface varies as some projects do both internally, others do both externally, and others split the responsibility. This strategy is based on the following four elements and some guiding principles:

(1) Governance matched to the need of the initiative. The intent is not to map back to NPR 7120.5 life cycles but to establish PDLM governance and life cycle in support of programs and projects.
(2) Requirements and capability-driven design.
(3) Appropriate levels of interoperability.
(4) Staging of technology.

Effective PDLM principles require aligning the organizational governance model and the technology, recognizing that:

a. Today's product data requires special handling due to its dependence on technologies, such as system modeling and 3D CAD.

b. Product data is created and usable in different forms (lightweight and native) by different communities across the full product life cycle.

c. It is good practice to minimize building custom solutions and customizing out-of-the-box software and rely on external standards and configurability.

d. Leverage institutional knowledge from PDLM implementers across NASA to provide a consistent approach to PDLM.

A properly implemented PDLM environment mitigates and manages program/project risks by:

a. Establishing a defined data management environment that provides a framework that ensures data security, data integrity, CM, secure user access, data classification, defined data interaction processes, and established standards for program/project data, including models-based data.

b. Identifying all the various life-cycle phases and product development stages and ensuring that the requirements for and governance of program/project data management are developed that encompass all of this.

c. Implementing an engineering and change process that addresses the different release definitions and data object use cases across operating boundaries and life-cycle states.

d. Establishing a parts library and management system with defined rules and processes that can handle special data types, particularly electrical computer-aided design (eCAD), mechanical computer-aided design (mCAD), and computer-aided manufacturing (CAM).

e. Addressing data interoperability and ensuring that contract language supports the level of interoperability required to enable program/project success.

4.1 Governance

The nature of required decisions varies during the development and life of a program/project, and governance should reflect that. The following should be considered for a governance model in which participation varies by the phase of the PDLM effort from Formulation to Planning, Implementation, and Sustainment, commensurate with the NASA program/project life cycles stated in figure 1:

a. Early Governance: support focuses on participants with understanding of both the future vision and the existing barriers to achieving it.

b. Later Phase Governance: will provide a predictable path for new capability requests and upgrades that preserve committed functionality. Transition between phases will be part of the planning, coordination, and management to be conducted and captured in the PDLM Plan.

c. Formulation Phase: During the project Formulation Phase, governance is often less formal as the early advocates search the existing organization to build a well-formed vision statement and develop needs, goals, and objectives statements. Key personnel should be selected who can provide critical guidance and help build a team to coordinate and execute the later phases. Governance mechanisms in the Formulation Phase should reflect the known challenges in the PDLM domain and provide a means to capture and preserve Formulation plans, concepts, alternatives, and any associated meta-information. The goal of governance in this phase is to enable communication and preserve the resultant complex data without impeding the development of the needs, goals, and objectives.

d. Planning Phase: During this phase, representatives from Centers and other organizations who will be affected by the initiative are given a mechanism to develop and then formally review Project Plans and assess their impact. Governance mechanisms in the planning phase should reflect the known challenges in the PDLM domain and provide a means to coordinate the creation of detailed plans, including identifying potential conflicts and proposing alternatives. The goal of governance in this phase is to validate an achievable plan to meet the needs, goals, and objectives.

e. Implementation Phase: As the plans identify specific resources and assign actions and sequence, the governance should transition to focus on the management of the implementation details. The ability to match priorities and resources is required with major issues that appear to impede the implementation receiving executive-level attention.

f. Sustaining Phase: As the body of the implementation is deployed, the governance focus shifts to a model in which existing capabilities are reliably delivered. Sustaining governance models provide a defined path for existing users to request changes, confidence in predictions about when new services and functions will be provided, and an approach that emphasizes cost justification.

This mapping of governance-to-program/project phases is an oversimplification—execution will require the chartering of committees, forming of temporary working groups, and creating communication pathways between them.

Answering the question about who decides what and when will depend on decisions to be made about the scope, scale, and timing to move forward. Strategically, what is important about governance is that participation in decisions reflects the dependencies and impacts of changes in how product data are managed across the extended program/project life cycle.

4.2 Technology

It is important to implement PDLM solutions that use COTS software and Web technology and are built to be extensible and scalable to address new and changing requirements.

Critical items to be considered include:

a. Designing and implementing technical solutions for PDLM require resources and knowledge.

b. Even the simplest consolidation of users onto an existing system requires technical support for the creation of accounts and the transfer of data, user training, and coordination with customers, their managers, and the technical support team.

c. Consolidations that involve greater technical effort would require the creation of new objects and the redefinition in the new location (or partition of an existing implementation), in addition to the simplest consolidation.

PDLM technology is often procured as a single integrated tool suite provided by a single vendor comprised of commercial database management systems; custom-written software; off-the-shelf interface tools; report writers; integration and utilities elements; and configuration, auditing, administration, and definition files. While the value and economies of such a solution are evident, no one tool can accommodate all the needs in the heterogeneous environment present in many NASA programs/projects; and the evaluation and selection of the correct technology for PDLM is critical.

5. PDLM FRAMEWORK

The PDLM framework is the distributed and federated data management environment that creates the baseline from which all program/project data management features and deliverables will be driven. At the core of this framework is the infrastructure on which all collaboration, workflows, data management, and data sharing will be built. Implementation of PDLM requires the establishment of an Agency-level collaborative environment (defined by NPR 7120.9 as four architecture types: Security, Data, Information Support System, and Process). The program manager and Technical Authority are responsible for documenting in the PDLM Plan the scope of each Center's responsibilities; the implementation approach; the environment in which the program and its projects' PDLM solution(s) operates; and timing for implementation of the PDLM solution and associated processes, data, elements, attributes, and formats.

A defined PDLM framework is needed because, if the relationships linking supporting documents and the product structure are not defined, searching for data becomes time-consuming and cumbersome. For example, without upfront planning, part numbers and descriptions become inconsistent, incomplete, and consumed with search parameters; renaming CAD files (due to conflict) rather than replacing them causes non-optimal data proliferation; and scalability is becoming a bigger concern for the suppliers of the tools and applications due to constant and sporadic increases in size and complexity through the lifetime of the programs/projects. For programs/projects operating across multiple Centers and suppliers, processes may vary greatly and translation may become time-consuming, cumbersome, and costly.

An established PDLM framework ensures the following:

a. Security: All PDD is captured through a secure environment that meets the data's security requirements as defined by NPR 2810.1, Security of Information Technology, regardless of the data's form (digital or hardcopy).

b. Product Data Integrity: Relationships between all associated product documents are defined and managed.

c. Automation: Workflow and automatic notification capabilities are provided.

This planned environment ensures that all product data (CAD and product documents) are captured and distributed. Relationships between all the associated CAD parts and product documents are defined and managed. Data, particularly CAD part numbers, are searchable by attributes and metadata. Product data are controlled as defined by contracts and data sensitivity and are securely shared as needed and when needed.

5.1 Security Architecture

The Security Architecture exists to ensure continuity of Mission Directorate operations for both flight projects and ground operations through the management of risks and vulnerabilities to protect humans, flight and ground facilities, information, systems, and services against disasters, threats, errors, and manipulation. The Security Architecture is a collection of components or layers of security that provides information assurance. These include policy and security management, application security, data security, platform security, network and perimeter security, physical security, and user identity security. It provides the program/project with reliability, quality, integrity, availability, and confidentiality of data and systems in compliance with Federal and Agency regulations and requirements. It is the responsibility of the Agency Chief Information Officer (CIO) to develop the Security Architecture as defined in NPR 2800.1, Managing Information Technology, and NPR 2810.1. The NASA authorizing officials ensure that appropriate IT elements and approved IT security plans are in place and address the required elements as required by their Center-level procedural requirements documents.

5.2 Information Support System Architecture (ISSA)

The ISSA is comprised of IT components allowing users access to data and information under a configuration-managed, secure environment. The ISSA allows programs/projects to capture, integrate, and manage product and process information from diverse authoring applications in a single environment. This environment enables the definition and standardization of workflow-driven processes that can be leveraged and used across multiple programs/projects. The ISSA is designed to integrate multiple mission-critical systems through the use of industry standards in PDLM (e.g., open APIs and enterprise service buses) to aggregate and extend knowledge sharing throughout the organization. It is the responsibility of the Agency CIO to develop the ISSA. The Agency CIO ensures that IT and information resources are acquired and managed in a manner that implements the policies, procedures, and priorities defined by NPR 2800.1 and ensures that IT products, services, and information systems meet customer and stakeholder requirements.

5.3 Data Architecture

The Data Architecture is a representation of data artifacts and data assets that classifies and defines all data entities, their attributes, and associations to facilitate knowledge of how data is produced, managed, and shared in different contexts of use. A Data Architecture describes how data is processed, stored, and utilized in a given system or product definition. It provides criteria for data processing operations that make it possible to design data flows and also control the flow and association of data in the system. If programs/projects do not have the Data Architecture from the start, the program/project data becomes disorganized quickly and is nearly impossible to electronically integrate or discover as the level of data increases over time.

The Data Architecture provides for creation, storage, and exchange of data, especially PDD, models and simulations, and for the requirements imposed on the PDLM solution elements to support data interoperability, management, integration, metadata, and work practices, including, but not limited to, standardized taxonomies and ontologies, and should be based on common, open standards or NASA standards.

The Data Architecture processes and requirements have to be flexible to accommodate changes in PDLM vendors, NASA Center and institutional software, and environments and allow the inclusion of external suppliers or partners that use their own local PDLM environment or do not currently support a PDLM environment.

The CIO provides the infrastructure/systems for implementing an architecture that supports the creation, storage, interoperability, and exchange of data. Engineering defines the program/project data that the architecture has to support and defines the criteria for data processing operations, data flows, and the association of data in the system. Further, engineering supplies the requirements for the infrastructure/systems needed to meet these criteria. The program/project manager and Technical Authority are responsible for ensuring that the Data Architecture meets the program's/project's needs. The program/project Data Architecture should be defined and documented in the PDLM Plan.

5.4 Process Architecture

Processes enable the integration of diverse human and systems activities to coordinate the exchange of services and data/information, including process design, process execution, and process monitoring. A Process Architecture is a description of the program/project business processes. The Process Architecture establishes standards in how the program/project will create, manage, and control program/project data. A Process Architecture describes data interoperability requirements at defined life-cycle states and maturity levels (both internally and externally to NASA) and which processes will be applied across all projects (e.g., safety and mission assurance, deviations and waivers, corrective actions, and problem resolution) and identifies the source for documented common nomenclatures and schemes for product naming, numbering, and version control.

The Process Architecture defines flow of the program/project data, the roles and responsibilities for data handling, and the approval authority. The Process Architecture should clearly define, control, and integrate business and engineering processes for program/project execution based on results from life-cycle cost analysis. Processes should be subjected to value-added analysis techniques (e.g., streamlined, efficient, effective processes) prior to implementation. A well-defined Process Architecture will ensure that every person or organization involved executes the process. Each process to be used is fully documented so that everyone understands their respective involvement in the process.

The program/project manager and Technical Authority are responsible for defining the relationship between business and engineering processes and documenting these relationships in the PDLM Plan. The Agency CIO is responsible for implementing the requirements of the Process Architecture.

If established early and properly, the key benefit of the Process Architecture is that all program/project team members will clearly understand data control and release, including which data are authoritative and the process through which the data are released to become authoritative.

5.5 Models-Based Design

Data through all life-cycle phases can cross many entities that will likely change over time. That data today is generally electronic, the design environment continues to move toward a models-based approach[1], and the engineering/acquisition processes are more automated and demand a highly collaborative/social workforce during all phases of the life cycle. It continues to become more important to ensure the following are considered:

a. Design capture—schematics, analysis, CAD, eCAD, CAM, 3-D scan, surveys, field measurements, simulations, etc.

b. Manufacturing and assembly—as designed/as built.

c. Safety and Mission Assurance (S&MA)—problem reporting and corrective action, hardware screening, reliability, failure mode and effects analysis/critical items lists, etc.

d. Operations—system is on-orbit operational; mission operations needs access to the authoritative data.

e. Management—needs access to dashboard to see if any trends were previously identified.

f. Ability to insert new software modules—Data Architecture allows for software upgrades while maintaining configuration.

g. Common data schema- and standards-based-driven requirements.

[1] Models-based approach: An approach to developmental engineering in which the design and its associated analyses, specifications, etc., are created, used, and managed as a non-document digital data object (or multiple linked objects); different formats exist to support different engineering domains and these divergent needs and data differences impact interoperability, data integrity, and engineering data management.

h. CM of the models to ensure data integrity and model validation.

i. Risk.

In model-based design, a system model is at the center of the development process, from requirements development, through design, implementation, and testing. The model is an executable specification that is continually refined throughout the development process. After model development, simulation shows whether the model works correctly.

The true benefit of a models-based environment may not be seen at the Center or authoring level; however, upstream and downstream users are hugely impacted as the amount of work required to validate or modify the completeness or correctness of the data being shared increases. Furthermore, a models-based environment as part of a PDLM strategy also covers the integrity (security, relationships, revision currency, effectivities, standards compliance, etc.) of the data.

The following table 1, CAD Interoperability Insights for Project Data Acquisition and Exchange, illustrates use of data in a models-based environment, in particular for 3D CAD interoperability and how to facilitate better planning and acquisition of design data.

Table 1—CAD Interoperability Insights for Project Data Acquisition and Exchange

If we want to:	Likely, we would need CAD specified like this:
Do design integrations	Pre-release models with data appropriate for the life-cycle phase to defined (high) accuracy based on negotiated agreement between provider and user.
Subcontract part of design work	To send contractor skeleton files, start parts, approved parts lists, etc., and receive back from them design files, component models, and other elements of the product definition package.
Take over design change authority	Full design history and 3D CAD models for a given vehicle block, including skeletons, standard parts, start parts, drawings, configuration settings, model-checking criteria.
Conduct design review	3D “viewable” of released models or external alternative representations that support annotations.
Do derivative designs such as GSE	Pre-release CAD models from source at defined maturity level treated as “released” by users within their own environment with associated metadata appropriate for the life-cycle phase.
Do modeling and simulations	Pre-release alternative representations with full definition of items significant for modeling and simulation with data appropriate for the life-cycle phase.
Re-bid production	Released CAD models, drawings, material specifications, manufacturing processes, installation models, etc.
Do physical integration and verification (e.g., at Vehicle Assembly Building)	Released alternative representations or source CAD with appropriate alternative representation substitutions with full definition of items at integration point (but lacking internal detail) with data appropriate for the life-cycle phase.

If we want to:	Likely, we would need CAD specified like this:
Define contents of Acceptance Data Package	CAD need based on defining requirements—who/what/when/how models will be used—to ensure that they are the right life-cycle/maturity state, format, and contain needed content.
Perform Concept Development and Refinement	Surface, shape, parametric, possibly detail design or as-is (for existing systems) and "will-be" (for future systems) model and simulation data.

If the PDLM environment is used throughout the life cycle, work products can be built and matured seamlessly, eliminating the need to re-create them over the life cycle, thus resulting in enhanced quality and affordability. Example products are shown in table 2, Example Products in the PDLM Environment.

Table 2—Example Products in the PDLM Environment

Pre-Phase A	Phase A	Phase B	Phase C	Phase D	Phase E	Phase F
Concept Studies	Concept & Tech Dev.	Prelim. Design	Final Design & Fab.	Assembly, Test, & Launch	Ops & Sustainment	Closeout
Conceptual Models and Simulations, especially MBSE Cost Estimation	Requirements Functional Flows Models and Simulations, especially MBSE	CAD Designs Analysis Models Prototype Test Data Refined Costs Models and Simulations, especially MBSE	Models and Simulations, especially MBSE GD&T PMI Inspection Data Change Orders Effectivities	Integration Models and Simulations Verification Certification Change Orders Effectivities	Operations Anomalies Simulations Science Data Change Orders Effectivities	Decommissioning Simulations Data Archiving Final Costs

5.5.1 Models and Simulations (M&S)

M&S are used daily for making critical decisions in design, development, manufacturing, ground operations, and flight operations. The program/project manager and Technical Authority are responsible for ensuring the credibility of the results from these M&S activities. Confidence in modeling and simulation results is achieved by:

a. Identification of best practices to ensure that knowledge of operations is captured in the user interfaces (e.g., users are not able to enter parameters that are out of bounds).

b. Development of processes for tool verification and validation, certification, re-verification, revalidation, and recertification based on operational data and trending.

c. Development/identification of standards for documentation, CM, and quality assurance.

d. Identification of any training or certification requirements to ensure proper operational capabilities.

e. Development/identification of a plan for tool management, maintenance, and obsolescence consistent with modeling/simulation environments and the aging or changing of the modeled platform or system.

f. Development of a process for user feedback when results appear unrealistic or defy explanation.

g. Development of a standard method to assess the credibility of M&S presented to the decision maker when making critical decisions (i.e., decisions that affect human safety or mission success) using results from M&S.

h. Assurance that the credibility of M&S meets the project requirements, and the appropriate documentation is captured and preserved.

NASA-STD-7009, Standard for Models and Simulations, provides requirements and recommendations for achieving these eight objectives. NPR 7120.9 requires that the PDLM Plan address how the elements of NASA-STD-7009 are applied to product data management. NASA-STD-7009 does not apply to M&S that are embedded in control software, emulation software, and simulation environments. For these activities, the Center implementation plans for NPR 7150.2, NASA Software Engineering Requirements, should cover embedded M&S and M&S-specific issues.

5.5.2 Models-Based Systems Engineering (MBSE)

MBSE is the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life-cycle phases[2].

The intent of MBSE is to facilitate traditional systems engineering resulting in enhanced communications, specification and design, system design integration, and reuse of system artifacts and models. Currently, MBSE supports engineering data but not business data. It is expected that in the future MBSE will extend to domains beyond engineering to support complex predictive and affects-based modeling that includes integration of engineering models with scientific and phenomenology models; social, economic, and political models; and human behavioral models[3]. The output of MBSE is a system model sometimes developed in Systems Modeling Language™ (SysML™).

---

[2] International Council on Systems Engineering (INCOSE), *Systems Engineering Vision 2020*, Version 2.03, TP-2004-004-02, September 2007

[3] MBSE State of Practice -2020. Presented at the INCOSE 2007 Symposium. http://www.incose.org/enchantment/docs/07docs/07jul_4mbseroadmap.pdf

MBSE helps teams focus thinking to develop engineering work products. MBSE captures the decisions, performs analyses, and produces review and audit materials. An MBSE system model can incorporate models to document critical design sets and graphical presentations of process flows that enable group participation in concurrent engineering. This includes CAD models, digital manufacturing models, as-built models, model-based testing, and logistics functions.

MBSE is a method to drive models that represent functions, requirements, and conceptual systems and is present and interactive within the life cycle (design through retirement). PDLM is the management of product data across the product development cycle and applies a consistent set of business solutions that support MBSE. PDLM is the underlying architecture required to tie data and processes together and provides the framework for collaboration and interoperability. PDLM and MBSE are complementary and integrated efforts. PDLM provides the capability to orchestrate procedural events such as design reviews from authoritative data sources. MBSE provides the capability for these reviews to be conducted with a single-source system model that provides viewing of the system model from different perspectives (i.e., discipline-dependent views).

A listing of MBSE resources is included in Appendix A of this Handbook.

6. PDLM ELEMENTS

6.1 Requirements Management

Requirements management in a PDLM program/project environment provides a framework for defining, refining, documenting, and maintaining the requirements of the product(s) produced by the program/project. The requirements are to be maintained and managed throughout the entire life cycle of the project. Program/project requirements management is defined and documented by the program's/project's systems engineering and integration functions or their designees and documented in the Systems Engineering Management Plan in accordance with NPR 7123.1.

The size (scope and scale) of a program/project has an impact on the PDLM solution but is not deterministic in regard to requirements for managing product data. Factors such as the nature of the mission, mission class, the amount and types of data, data acquisition, sensitivity, retention needs, the location, timing, and duration of data access, and the organizational relationships of participants are important for characterizing the demands that programs/projects make on PDLM capabilities.

This element provides an authoritative set of PDLM requirements that represents the various life-cycle phases of the products and the current architecture of the product at all stages. PDLM requirements management should:

a. Allocate requirements to various program/project organizations for decomposition.

b. Define the relationship of requirements to other information in the PDLM environment.

c. Provide the framework for requirement derivation and a scheme for parent and child requirement relationships.

d. Provide links for access to authoritative requirement information.

e. Allow the definition and management of detailed information attached to the requirement such as margins, technology readiness levels, risk, and cost data.

f. Be traceable to its source, downstream to its implementing product component, and to their verification and validation activity.

g. Be securely controlled and traceable from beginning to end of life (history and use).

h. Be captured in a virtual database that provides secure but easy access. Requirements management tools should provide automatic notification of requirements changes and impact to flow-down or linked requirements.

i. Allow simultaneous management of multiple system versions, i.e., Orion 602 and Orion 604, to allow for management of both existing baseline systems, as well as projected alternatives.

6.2 Configuration Management

CM in a PDLM environment provides a framework for programs/projects to develop and control their engineering documentation and data that define the following:

a. Functional and performance characteristics of the program's/project's products.

b. Functional, allocated, and product baselines.

c. Requirements for the design, manufacturing, testing, checkout, training, maintenance, operations, and disposal of the products.

d. Verification that the end product is in compliance with the requirements and associated information.

e. Traceability of waiver, deviation, and non-compliance of requirements information.

f. Retention, security, and data integrity of records.

g. The determined level of CM by supporting organizations.

The program/project-level CM processes establish and maintain consistency of products with requirements through the entire program/project life cycle via the PDLM environment. CM creates a consistent and systematic method for products delivered to the programs/projects that:

a. Identify the product configuration (architecture, current status, and related information) at the key decision points and other times.

b. Systematically control changes to the product configuration.

c. Maintain the integrity and traceability of the product configuration throughout the product life cycle (for example, operations anomaly functions may require fast access to information from the earliest product phases).

d. Preserve records throughout the product life cycle and properly retire product records.

The CM administrative function as defined in NASA-STD-0005, NASA Configuration Management (CM) Standard, shows how the product data should be released. Once data has been released, it follows the procedures and processes established by the program's/project's CM and Data Management Plans.

The program/project manager and Technical Authority define and document as part of the PDLM Plan what product data is released, when that product data is to be released, the events that necessitate product data change, and the processes that provide the visibility of the product data configuration life cycle for internally and externally produced product data.

6.2.1 Engineering Release and Change Management

Engineering release is defined as an action whereby engineering documentation or an item is officially made available for the intended use. All engineering releases, changes, and supporting documentation are properly managed and the terminology of the engineering release process properly defined. The change process should be a closed loop process with the changes automatically routed and the impact of the change easily visible. A history of changes should be maintained.

Program/project managers, Technical Authorities, and appropriate engineering organizations consider the following as part of the development of PDLM engineering release and change management:

a. Engineering release of data occurs during the following life-cycle states/phases:

(1) Requirements definition.
(2) Design and development.
(3) Fabrication/manufacturing.
(4) Test and verification.
(5) Flight and operational data.

The meaning of the term "release" can change during these different life-cycle states/phases. The term "release" is used to mean "you can do X with Y" where both X and Y vary. For example:

(1) Released Upper Stage CAD model of Liquid Oxygen (LOX) tank (Y) means the prime can start manufacturing (X).

(2) Released trade study of material choices for Upper Stage LOX tank (Y’) means designers can proceed with Upper Stage tank design work (X’)

b. Data objects are defined by the object itself and by its state (e.g., “in work,” “under review,” “released”). Additional data are required to indicate how the data objects relate to other data (e.g., effectivity). A method for capturing the object and relational data is to be developed.

The object type does not determine the meaning of state. For example: A CAD model can be released as:

(1) Design definition for manufacturing (e.g., Upper Stage LOX Tank), or

(2) Intended for use in outer mold line development.

c. Process detail depends on both object type and desired use and is not exclusively defined by project phase, e.g., more design work is done in the detail design phase, but design work is also done in operations to support block changes and sustaining engineering. Object type and usage are considered along with project phases during process development.

d. The program/project has to have confidence that data in a given state means the same thing to everyone:

(1) A process that produces known object in a known state is to be provided. For example: Product definition package (3D CAD, 2D model, and material specification) “released for manufacturing” always means released for production activities such as manufacturing planning, computer-controlled manufacturing machine programming, procurement, inventory management, etc.

(2) If work is distributed among Centers or suppliers, the program/project needs to assess what release means across context.

e. Different objects have different uses and users, and it is those use cases that should determine the meaning of the states (preliminary release):

(1) PDLM and CM distinguish between types of objects and their uses. For example: Several external suppliers use two design paths: prototype and manufacturing. 3D CAD models that are released for prototyping are NOT released for manufacturing.

(2) Within each use case and for each product data category (i.e., CAD, parts assembly, analysis, etc.), the program/project manager and Technical Authority define the life-cycle states (in work, approved, released, etc.).

f. The program/project manager, Technical Authority, and engineering determine if some data needs special handling and what standards should be used or developed to address this data. Examples of these data objects/types are engineering models/design/analyses (e.g., SysML™, 3D CAD/computer-aided engineering (CAE), etc.). (Another data type, for example, could be MATLAB designs used for modeling of guidance, navigation, and control (GN&C) algorithms and then has source code auto-generated for integration into flight software.

g. System synchronization, the process of collecting a known set of configurations for a system-wide test at various points in the overall system life cycle, enables early integration testing, as well as early investigation of alternatives.

h. With large amounts of externally provided data, it is difficult for NASA programs/projects, due to cost and other factors, to insist that suppliers (e.g., primes, partners) all follow a defined, Center-specific process. For this case, the program/project can and should insist that processes have key states that map to defined program/project core states:

(1) The PDLM Plan should provide a mapping of source states to destination states that provides confidence that the data can and will be used for the intended purposes and not unintentionally misused. For example, an external supplier can call it "manufacturing release," one NASA Center can call it "final release," and another NASA Center can call it "engineering release."

(2) Source-to-destination mapping confirms that all these are the same as the program/project-defined "production release." Refer to figure 3, Notional State Compatibility Map.

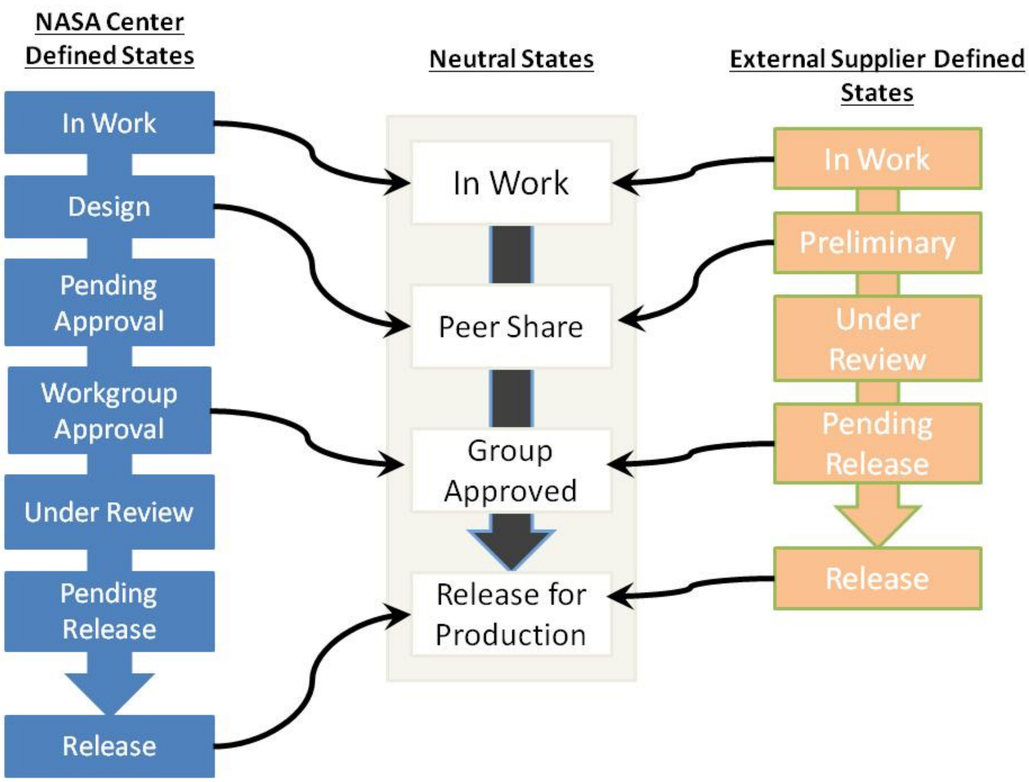


Figure 3—Notional State Compatibility Map

6.3 Risk Management and Mitigation

Each program/project is to be aware of and address the impacts of inadequate product data management occurring across the product's life cycle. The risks emerge from cases that can occur repeatedly at different links in the chain of data hand-offs among the multiple parties, thus ultimately impacting Certification of Flight Readiness:

a. Costs due to re-creating or converting data, designing, creating, testing, and operating custom data exchange mechanisms, revising contracts, and to rework caused by incorrect data, including repeated analyses, having to remake or re-transport parts or ground support equipment (GSE), and delays in starting or completing downstream activities due to any of the above.

b. Delays due to re-creating or converting data, arranging mechanisms of data exchange and managing data received and sent, revising contracts, and reworking dependent designs, analyses, or simulations upon discovery that the data used were incorrect.

c. Mission success risk is related to two items:

(1) The risk that the wrong data was used at some point along the chain, that the error was undetected, and that its consequences were severe enough to shorten or end the mission; and

(2) The time and effort associated with discovering data needed to support a flight anomaly, and the risk that that delay will shorten or end the mission.

Understanding how a program/project is impacted by the risks associated with program/project data management and how the PDLM environment addresses those risks are critical to program/project success. Having a fully interoperable system is insufficient if it cannot support peak user activity levels, meet system performance requirements, or if it provides access to the wrong data but represents it as an authoritative source. PDLM has to address the linking of user, function, infrastructure, and technology.

Programs/projects, in the development of their PDLM Plan, need to determine and assess their risk areas related to product data management, and then they need to ensure that the systems and architecture that support the program/project will mitigate these risks.

6.4 Parts Management

Parts data handling is distributed across Centers and out to primes, assigned to multiple different functional areas within projects, and maintained within locally controlled design environments. Effectively handling how parts are identified, defined, categorized, and authorized, and specifically, how CAD data objects are handled in systems for design and sustaining engineering is critical to performing the job of systems integrator. When part numbers exist without exact specifications, or exist with a part number that is similar to the part number of another component, time and resources can be required at a critical point in product development to reconcile the discrepancy. Product information has to be shared across product development boundaries and throughout the life cycle; but without everyone involved having an understanding of the specification and characteristics associated with a part number, problems will exist. This problem is not solved by assigning new part numbers. Part number proliferation causes confusion and duplication—not only duplication in effort but duplication in cost for maintaining parts inventory. Parts data handling processes should be defined upfront and should be enforced and followed for such matters as part naming and substitutes/alternatives.

Establishing parts libraries is critical in program/project development. Multiple libraries can exist such as part definition libraries, M&S libraries, and CAD libraries. Maintenance of these libraries can be spread across multiple organizations, internal or external to the program/project, and usage can be shared by many users across organizational boundaries. It is imperative that the program/project understands and documents the library structure and ensures approved practices are established.

6.4.1 PDLM Environment

The well-defined PDLM environment should:

a. Host the library used for CAD design.

b. Provide a distinguished security and access model for the libraries that restricts all changes to a designated set of librarians.

c. Provide efficient mechanisms for managing large families of similar items.

d. Prevent or reduce the occurrence of duplicate or multiple part numbers and designations for an identical physical device. A part number should map to a specific, unique set of specifications.

e. Address the maintenance of additional metadata for standard parts such as standard identifiers, extensions for tracking the configuration of mission sets, and effectivities which map revision/versions to the extensions.

f. Manage the parts data developed by the design organizations developing the product.

g. Allow for inclusion of new libraries and mechanisms as new organizations enter the environment, new suppliers are added, and parts suppliers are consolidated or removed from the marketplace.

h. Provide for library availability for the required life cycle of the program/project.

i. Host the library used for requirements development and requirement-to-part associations.

6.4.2 CAD Parts Library

This section specifically addresses CAD library development, but the methodology is applicable to other parts libraries.

A CAD parts library functioning in a PDLM environment provides an efficient and flexible framework for developing, promoting, using, and maintaining CAD common parts libraries, registries, catalogs, and associated support mechanisms. The description of these libraries, registries, catalogs, and databases may include terminology such as "standard," "preferred," and “electronic, electrical, and electromechanical (EEE).” The CAD functions supported by the libraries may include eCAD, mCAD, and CAM.

A CAD parts library’s functions should include the following:

a. Accommodate the use of multiple CAD common parts libraries by the organizations within the enterprise while requiring appropriate identification, naming, and usage. Establish processes to minimize the number of duplicate labels for the same physical part.

b. Provide a mechanism for a CAD developer or integrator to determine if a part is already defined in a common library within the enterprise's PDLM environment.

c. Establish rules and business processes to accept individual new part definitions and designations into the program’s/project’s common parts catalog, registries, and naming conventions.

d. Establish rules and business processes to accept wholesale existing part definitions and designations (entire existing parts libraries) into the program's/project's common parts catalog, registries, and naming conventions.

e. Establish security rules and business processes for access to CAD parts that are proprietary, intellectual property, or designated as sensitive but unclassified (SBU).

f. Establish and support audit and compliance functions to verify that naming conventions, configuration and data management, and data architecture requirements have been satisfied.

g. Establish rules and business procedures for external organizations that are not part of the program's/project's PDLM system to verify the uniqueness of proposed part names and enter new parts into the program's/project's parts catalog system (delivery from an external source).

The scope of the CAD common parts library interoperability processes should be flexible. The business processes for the CAD common parts library interoperability should be defined in the program's/project's CAD standards, CM, and PDLM documentation.

The PDLM Plan should describe how CAD standards will be applied over the program's/project's life cycle, along with use of other internal or external standards, practices, settings, and supporting tools with responsible parties. The plan should also identify the program/project data or documents that the CAD producer is to provide in addition to the CAD object to ensure full PDD such as parts lists, materials specifications, and acceptance testing specifications and where this material will be maintained.

6.5 CAD Data Management

Effective CAD data management enables immediate access to needed PDD, defines relationships between all associated CAD parts and assemblies, provides management of mCAD, eCAD, and CAM parts and assemblies, and provides management of systems and software across the entire program/project life cycle.

Problems occur with managing CAD data because:

a. Not all CAD objects come with CAD data packages—incomplete information causes iterations.

b. Conflicts over CAD naming—may have multiple names for the same parts.

c. Name may change—no way to see if it is the same geometry.

d. CAD drawings are converted to Portable Document Format (PDF[4]) manually and to legacy formats, e.g., as required for import into CAE tools.

e. CAD is moved manually—links and relationships may be broken.

f. Not all documents are version/revision controlled.

g. Inconsistent modeling standards.

h. No design intent captured or shared.

Programs/projects should work to define and implement a CAD data management environment where:

a. MCAD (parts and assembly) data are captured in a virtual database.

b. Relationships between all associated mCAD parts and assemblies are defined.

c. Ability to rename to-be-determined (TBD) items prior to release into PDLM.

d. Structured check-in/checkout process and revision control ensure data integrity.

e. Workflow and automatic notification capabilities enable adoption of product development best practices.

f. ECAD and systems (parts, files, and assembly) data are available through a virtual database.

6.5.1 Exchanging and Distributing 3D Models

CAD systems are all moving toward 3D formats; and exchanging this 3D data within engineering departments, between primes, and with other divisions and departments is critical to program/project development. However, not all of these use the same CAD system or have a CAD system. When parts are developed in 3D, the data are initially stored in the original format of the software used to design the part. If this 3D CAD data are to be made available to people who do not have this software, a neutral 3D format is necessary. Refer to NIMA-RPT-004, Future Data Exchange for NASA Programs.

Which neutral 3D format should be selected depends upon a number of factors. The purpose of this section is to explain the use cases for 3D data exchange and the importance of understanding neutral data formats to interoperability. The diagrams and definitions in this section are taken from a published PROSTEP AG White Paper titled 3D Formats in the Field of Engineering – A

[4]Adobe Systems Incorporated, original developer and copyright owner of PDF, ceded control of the specification to the International Organization for Standardization (ISO) in July 2008. ISO is now in charge of publishing the specification for the current version and for updating and developing future versions.

Comparison.[5] It provides an overview of neutral formats and serves as an orientation guide for selecting the appropriate neutral 3D format shown in table 3, Suitability of Neutral 3D Formats for Individual Use Cases. See Appendix A for information pertaining to the 3D formats listed in table 3.

Consider these five use cases when establishing PDLM element and interoperability requirements. Understanding neutral format data usage allows for the appropriate tools to be selected in the most cost-efficient manner, efficient data exchange mechanisms to be established, and for everyone requiring data access to have that access at the required level. The use cases are defined as follows:

a. Viewing: If the use of a CAD system is not desired, the visualization of engineering data using 3D viewers comes into play in a number of different situations: the presentation of product data, the representation of 3D models for information purposes (e.g., for a design review or marketing), and the realistic representation in virtual reality systems. While the simple viewing of the geometry is sufficient in many cases, in other cases, metadata or product and manufacturing information (PMI) also needs to be displayed. The high-performance visualization of large assemblies or design spaces and neighboring geometries is often an important criterion. In cases such as these, it is especially important that simplified representations are used.

[5] This White Paper is available at http://www.us.pdfgenerator3d.com/nc/en/product/white-paper.html.

The most important needs for simplified representations are:

Table 3—Suitability of Neutral 3D Formats for Individual Use Cases

Use Case	STEP	3D XML	JT	3D PDF
Viewing	●	●●	●●●	●●●
Data Exchange	●●●	●	●●	●
DMU	●	●●	●●●	●
Documentation and Archiving	●●	●	●●	●●●
Portable PLM Document	●	●	●	●●●

Legend:

●●●	Highly Suitable
●●	Well Suitable
●	Suitable with Reservations

(1) The source system-independent representation of the model data (geometry and metadata) with the required level of detail.

(2) The ability to filter the product structure, e.g., using views or layers.

(3) The execution of simple measurements.

(4) Representation of PMI.

(5) The ability to represent textures and sources of light for applications in the area of virtual reality.

(6) The availability of easy-to-use, cross-system viewers.

b. Data Exchange: In development processes, it may be necessary to exchange exact geometry between different CAD systems. This is, for example, the case if a supplier is using a different CAD system than the one used by the manufacturer. Another common situation is that one of the development partners makes a change to the geometry after it has been exchanged. In this case, merely viewing the data is not enough. What is needed is a typical and frequently used modeling method involving design in context of existing geometry.

When exchanging exact geometry, additional information describing the product is often needed. In this context, a distinction is made between PMI and metadata. While metadata refers to descriptions that consist only of text, e.g., information about the author or the release status of a model, PMI is often added in the form of 3D annotations. This makes greater demands on the underlying data format and on the application doing the processing. This means that grouping and filter options are needed to ensure clarity.

Another frequent requirement is that the processing of PMI be possible subsequent to data exchange. The protection of intellectual property is also playing an increasingly important role in data exchange, and it should therefore be possible to take this aspect into consideration.

The most important requirements are:

(1) The transfer of exact geometry and the entire product structure.

(2) The transfer of metadata, as well as PMI annotations (depending on the concrete use case involved).

(3) Ensuring correlation between the original model and the target model.

c. Data Integrity: Programs/projects are formulated and implemented following various NASA directives through multi-disciplinary teams of engineers and other personnel, with systems engineers, program/project managers, and Technical Authorities working across the other engineering disciplines to achieve integrated systems on schedule and within budget throughout the program/project life cycle. (Refer to NIMA-RTP-002, Data Integrity in NASA Programs and Projects[6].) A primary use of this data is for making informed decisions, by individuals and teams, including decisions made at life-cycle reviews involving the evaluation of various aspects of the program/project. In each of the life-cycle phases and reviews, these multi-disciplinary teams work with a multitude of data related to the systems being developed, including requirements, verification requirements, test verification requirements, failure modes and effects, risks, hazards, and problem reports, among others, and are often linked to each other. In operations, the data becomes critical for determining the readiness of the systems, with the consequence of poor data integrity having potentially catastrophic effects, including loss of mission. Two major forms of data integrity have been identified as follows:

[6] The referenced white paper NIMA-RTP-002, Data Integrity in NASA Programs and Projects, traces requirements for data integrity to NASA Procedural Requirements and describes how data integrity is a critical factor for achieving integrity of the program's integrated design, as well as the readiness of the operational flight and ground data systems. It also provides data management lessons learned.

(1) Integrity of Data Relative to its Authoritative Source

A. Integrity of data relative to its authoritative source is defined by the field of information security as the property that data have not been changed, destroyed, or lost in an unauthorized or accidental manner. In the context of NASA programs/projects, the most common and actionable form of this is the integrity of data relative to its authoritative source across and within a program/project.

B. Within programs/projects, there exist authoritative sources for various types of data involving, in many cases, the use of various discipline-specific tools and varying instances of the same tool. To conduct integrated analyses and reviews using data from one or more disciplines or from one or more tools, data from authoritative sources needs to be integrated. The common, current practice is to manually copy data from its authoritative source into other tools, documents, or presentations. From the moment the data is copied out of its authoritative source, there is a risk that the data is out of date, as changes to data in the authoritative source are not automatically made in the manual copy; the integrity of the data from the perspective of the authoritative source may be compromised. Existence of data discrepancies presents a high probability program/project risk of incorrect decisions being made, inadequate integrity of the program/project integrated design being represented, or inadequate readiness of programs/projects and their associated systems being presented.

(2) Integrity of Data Relative to an Authoritative Model

A. Integrity of data relative to an authoritative model is defined by the field of database systems as the property that data is an accurate reflection of the domain of discourse it is attempting to model. In the context of NASA programs/projects, the most common and actionable form of this is the composite and component integrity of data relative to an authoritative model.

B. Authoritative models of valid forms of data exist, from a single piece of data relative to the model of a valid data type, range of data, or simple text rules (component data integrity), to more complex data involving multiple pieces of data relative to each other and an authoritative model (composite data integrity).

The primary solution for improved data integrity is the use of information systems to change from manual processes of data integration and validation to automated processes. Two main approaches, automated data integration and automated data audits, are addressed as follows:

(1) Automated Data Integration Relative to Authoritative Sources of Data

A. Goals of using this approach include improved data integrity and data accessibility and elevated efficiencies in data management over the full program/project life cycle.

B. Lack of data integration has been identified as leading to cost and risk increases due to use of non-authoritative/out-of-date data, miscommunication of data, and missing data. With this approach, use of non-authoritative copies of data is minimized through automated data aggregation. Consequently, rates of engineering data discrepancies are also reduced whenever automated data aggregation is used. User communities can manage specific data in tools explicitly designed for the management of that data and can also implement specific processes that integrate data directly from the authoritative sources into reports and other formats.

(2) Automated Data Audits

A. Performing automated audits using information systems is a proven approach that has high utility and relatively low cost.

B. In the engineering realm, data audits can be performed daily or weekly to help drive convergence in design, and can serve to ensure the integrity of the data that is used in baseline versions of a multitude of engineering artifacts. This can be done at the elemental level with a single piece of data, or at the composite level with multiple pieces of data.

d. Digital Mock-Up (DMU): In DMU (computer-aided test model), the mechanical properties of a product are examined and checked. This can involve checking the overall geometry with regard to dimensions and shape, interference checks, and collision checks for assembly and disassembly, as well as design space checks.

For these purposes, the geometry, product structure, and metadata are displayed and analyzed in a DMU application. A distinction is made between static and dynamic DMU analysis. In the case of static DMU, an examination of the static parts is performed. In the case of dynamic DMU, the dynamic parts or assemblies are examined.

As a general rule, a simplified, tessellated representation of the envelope geometry is usually sufficient for use in DMU. In the case of measurements, however, it should be noted that the level of tessellation accuracy has to always be higher than the required measuring accuracy.

The most important requirements are:

(1) The availability of applications that support the respective required DMU functionality (e.g., assembly checks and collision control).

(2) Use of models from different source systems (multi-CAD).

(3) High-quality examination of large assemblies.

(4) Transferability of kinematics from the original model to the target model for dynamic DMU analysis.

e. Documentation and Archiving: For the purpose of documenting and archiving engineering data, it is normally necessary to factor in exact data representation, including all metadata and PMI. The latter is especially important with regard to product approval and product documentation if drawings and technical documents are being replaced by digital 3D models. There are also often compliance requirements that need to be satisfied.

The most important requirement relating to the documentation and archiving of engineering data is that all relevant information be stored in a format that can be read irrespective of a specific IT infrastructure and after a long period of time; in the aerospace industry, for example, the Long Term Archiving and Retrieval (LOTAR) activity (http://www.lotar-international.org/lotar-standard.html) that addresses archiving of data for long life-cycle programs.

The most important requirements are:

(1) Ensuring proper consideration is given to all product data.

(2) Ensuring problem-free combination of data from different source systems.

(3) Ensuring that the data can be accessed even after long periods of time (standardized format)

f. Portable PLM Document: In modern product development processes, the utilization of 3D information does not end in the engineering department. Integrated product development requires that departments such as purchasing, quality assurance, technical documentation, planning, and manufacturing also have to be able to access 3D information in combination with a wide variety of different documents such as bids and requests for quotes, quality control checklists, technical and manufacturing documentation, etc. It is important that this content can be combined in a multimedia container that includes all the information and can therefore also be used offline.

The most important requirements are:

(1) Information in the form of 3D data, metadata such as 2D representation, text data, and binary data can be combined in a single file and can be managed there.

(2) The data can be combined easily with information from various source systems.

(3) Comprehensive control options for file access (Intellectual Property Protection) exist.

(4) Easy to use, cross-system viewers are available.

6.6 Product Breakdown Structure (Bill of Material (BOM))

A product breakdown structure is a hierarchical view of the relationships of products and component products. Product structure provides a platform on which an entire program/project product family can be based. Product structure involves decomposition of the overall functionality of a product into a set of defined functions and the component parts of the product that provides those functions and the specification of the interface between the components—how components interact together in the product as a system. It is a product data structure that captures the end products, its assemblies, their quantities, and relationships.

A BOM is a subset of the product structure that includes the physical items and is a formally structured list for an object (semi-finished or finished product) that lists all the component parts of the object with the name, reference number, quantity, and unit of measure of each component. PDLM should provide for traceable documentation of all changes to the BOM, including when and why it was changed, which documentation implemented the change, who approved the changes, and the disposition of the affected items.

Problems occur with the development and use of BOMs because:

a. Only the "as-designed" BOM is managed in engineering.

b. Proposal BOM is created, then not carried forward.

c. Not all data delivered is related to part definition.

d. No product structure is delivered.

e. Effectivity for BOM change is limited to a single change and by date only; most recent is all that is saved.

f. Substitutes and alternatives are managed as separate BOMs and products.

Programs/projects should work to define and implement a product structure and BOM development that can:

a. Manage product structure and BOM data.

b. Capture and manage substitutes and alternative parts.

c. Provide revision control and history of product (BOM) data.

d. Manage multiple views of the BOM, including as proposed, as designed, as built, and as launched.

e. Manage effectivity by event, date, lot, or serial number as necessary.

The product structure should be documented in the PDLM Plan and provide a simple, concise view of the product that can be used as the central point of reference for all information describing and related to each element, subsystem, assembly, subassembly, and part comprised within the structure. The complete product breakdown structure segment list should be defined, including design data for an assembly and/or subassembly such as, as designed, as built or as manufactured, as manifested, as flown, and as disposed, beginning either at the highest level (system) or lowest level (component) to access all related data based on access privileges.

A good product structure should:

a. Be a decomposition/interrogation of the system into its parts.

b. Define the complete segment list, including design data for an assembly and/or subassembly, such as:

(1) As designed—a product structure representing the configuration after release, before manufacturing is started.

(2) As built (as manufactured)—a product structure that represents the configuration as it has been manufactured and delivered to the NASA inventory.

(3) As manifested—a product structure representing a configuration designated for a specific flight.

(4) As flown—a product structure representing a configuration that has flown.

(5) As disposed—a product structure representing a configuration that has been decommissioned.

c. Provide the capability for an individual to start at the highest level (system) to access all related data (based on the individual's access privileges) by drilling down to the lowest component of the system, as well as walking up from the lowest component to the highest level, including, but not limited to, access to supporting design documents such as analysis reports, test reports, requirements, and verification and validation checklists.

Components include hardware, software, and configurable logic devices. Note example below:

- System
- Segment
- Element
- Subsystem
- Assembly
- Subassembly (component)
- Part (piece part)

6.7 Digital Data Standards and Contract Language

NASA personnel spend a significant amount of time and cost in the role of systems integrator, relying on contractors and partners to deliver their data in a specific format to a particular system. Contract offices have typically used boilerplate templates in Requests for Proposals' (RFPs') contract language to define data interoperability. Although data delivery requirements have progressed into the digital age, the requirements have not been modified to fully embrace the benefits of digital data. Definition and interoperability requirements should be written into program/project contracts to leverage a data-centric approach for digital asset data management. This section describes the importance of connecting the product and program objectives with appropriate contractual deliverables, e.g., supplier deliverables required per program Data Requirements Descriptions (DRD).

The program/project manager and Technical Authority identify and acquire essential contractor-originated data with sufficient access and usage rights to support the full program/project life cycle. Contractor data interoperability is achieved through identification of IT standards and guidance on contract language.

Digital data deliveries have primarily been represented by digital scans of documents or in PDF and stored in a file system or content management system. This approach to data interoperability has fallen short in that data delivered as part of a page-formatted document lacks the relevant context required to effectively leverage the data access and interoperability potential inherent to digital assets.

That is not to imply that exchanging digitalized data will solve the data interoperability problem. Even digital data replication from system to system presents problems without a computer-readable format that maintains the referential integrity of the data. As an example, experience has indicated that the manual replication of data from one application to another could typically create a discrepancy in 40 percent of the transferred data (refer to NIMA-RPT-004). The disposition of that amount of data, per transfer, increases the argument for standardization to bring about affordability and sustainability.

The need was recognized for NASA to adopt industry standards and approaches for digital data interoperability to enable data interoperability and move beyond the storage of raw, uncorrelated data. The following sections contain the contract language and the associated standards necessary to execute digital data interoperability, which benefits NASA and its contractors and partners.

Proper data interoperability capability is achieved through the identification of IT standards and the application of contractual methods to ensure implementation and alleviate the data interoperability limitations that currently challenge NASA's programs/projects. There are multiple options for data interoperability among NASA Centers, and programs/projects with contractors, industry, partners, academia, services, and platforms. This section provides information to ensure that solution maturity is aligned with organizational requirements, proper data formats are applied, and data transmission types are understood. In addition, guidance on contract language is included to enable successful preparation of an RFP on data interoperability.

Digital data standardization provides:

a. Consistency of data and data use across systems.

b. Minimal redundant data collection and entry.

c. Maximum utilization of relevant technology.

d. Improved audit trail accessibility. Also, rapid retrieval of this audit trail enables:

(1) Better informed, real-time management decisions concerning changes.
(2) Increased quality in CM and control.
(3) Support for effectiveness of analyses of metrics.
(4) Improved data acquisition times.
(5) Improved data review and integration for:

A. Elimination or reduction of outdated or discrepant data copied into non-authoritative sources.

B. Elimination of manual management of relationships between data.

Digital data standardization should be specified when:

a. The potential for reuse of data exists; reuse may not be the same as for reviews or other occasions to "revisit" the data.

b. A need exists for interoperability of standardized data among NASA Centers, and programs/projects with contractors, industry, partners, academia, services, and platforms.

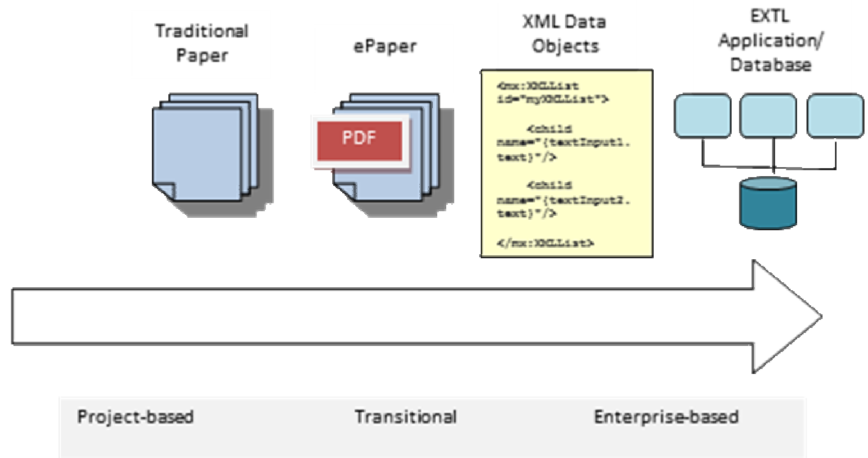
c. The need to provide relationships between data for performing program/project management, systems engineering, and operational functions exists.

6.7.1 Data Interoperability Formats

CAD data interoperability formats should be chosen based on the need for interoperability or reuse after delivery. Currently, MCAD data formats may be selected from table 3. In table 4, Format Maturity Levels, a maturity level is assigned to each data format by which the dynamics of each can be judged against intended utilization. (Refer to NIMA-RPT-004.) However, users should assess the current trend in technology from other sources prior to selecting data interoperability formats.

To build off the format and maturity levels of available technologies, the following guidelines are provided for when to apply the format:

Table 4—Format Maturity Levels



Maturity Level	Level 1	Level 2	Level 3	Level 4
Data Format Descriptor	Paper-based	ePaper	XML Data Object	Database w/EXTL
Delivery Mechanism	Physical Delivery	ZIP Electronic Delivery	ZIP Electronic Delivery	EXTL Application and Database Delivery
Standards Used	None	PDF	XML, XML Namespaces, XSD	The STEP Application Protocols (APs) (e.g., ISO 10303, AP203, AP209, AP214, AP233, or AP239)
Content Repository	File Cabinets	File Share to ECM	ECM to Structured Database	Structured Database
Recommended Utilization	Limited-life paperwork directly associated or co-located with hardware	Plans, papers, or other data delivered in association with hardware delivery/program management	Data intended to interact or affect other data within an enterprise environment	Data intended to interact or affect other data within an enterprise environment

Note: Intended Utilization Options

a. Level 1 (Paper): Documents delivered on the printed page. Specify when reuse and/or life-cycle maintenance of materials is not desired. Note: This approach is not recommended.

b. Level 2 (Formatted Documents)[7] [8]: Documents delivered electronically in a customer-approved, contractor format but containing no attached electronic semantic meaning. Specify when:

(1) Standard digital data can be delivered as electronic paper in a digital form. Electronic paper should include metadata tags, including, but not limited to, data type, system, title, and last modified to assist with document identification, storage, indexing, and key word tagging down to a minimum of three levels of subsections as defined by the program/project.

(2) Reuse and/or life-cycle maintenance of materials is not desired.

c. Level 3 (Standard XML Data Descriptions for Digital Data in NASA Standard Format): For standard XML digital data descriptions to be usable, a common data dictionary has to be referenced by all producers and consumers of digital data as defined by NASA's common data dictionary. If a NASA standard is not applicable, digital data has to reference a common data dictionary as defined by the contractor. A catalog of XML schema data descriptions should exist prior to the delivery of contractor data to NASA. Specify when:

(1) The potential for reuse of data exists.

(2) A need exists for interoperability of data described by XML schemas that move between/among NASA Centers, and programs/projects with contractors, industry, partners, academia, services, program/project life-cycle states, and/or platforms.

(3) Tagging data according to law, policy, and classification is needed at the data object level.

d. Level 4 (Digital Data from a Source Repository to a Target Repository in a Program/Project-specified Format)[9]: Digital data extraction from a source data repository in supplier format, data transformation into a target format, and data loading into a target data repository. Specify when:

[7] Zip is a file format used for data compression and archiving. A zip file contains one or more files that have been compressed, to reduce file size, or stored as is. The zip file format permits a number of compression algorithms. Zip files generally use the file extensions ".zip" or ".ZIP."

[8] Enterprise Content Management (ECM) consists of strategies, methods, and tools used to capture, manage, store, preserve, and deliver content and documents related to organizational processes.

[9] Extraction-Transformation-Loading (EXTL or sometimes ETL) is a process in database usage and especially in data warehousing that involves: Extracting data from outside sources, Transforming it to fit operational needs (which can include quality levels), and Loading it into the end target (database or data warehouse).

(1) The potential for reuse of data exists.

(2) A need exists for interoperability of standardized content between/among NASA Centers, and programs/projects with contractors, industry, partners, academia, services, program/project life-cycle states, and/or platforms.

(3) A content interoperability management system that conforms to an existing or planned set of NASA information systems and data capabilities and can receive the content.

(4) A need exists for interoperability of database repositories between/among NASA Centers, and programs/projects with contractors, industry, partners, academia, services, program/project life-cycle states, and/or platforms.

(5) A database-to-database integration approach exists.

(6) Tagging data according to law, policy, or classification is needed at the data object level.

6.7.2 Industry Standard Data Interoperability Formats

The following provides guidance for industry standard data interoperability formats that are well suited for relational and object model data interoperability implementations at the time of this document's release. For reference documents, see Appendix A.

The interoperability formats in table 5, Target Industry Standards for Data Interoperability Formats, are based on the World Wide Web Consortium (W3C[10]) Internet Standard: XML and supporting protocols. These standards are specific to the applicable data interoperability formats in maturity level 3 in table 4.

[10] World Wide Web Consortium, http://www.w3.org/

Table 5—Target Industry Standards for Data Interoperability Formats

Standard	Short Description	Industry Organization
[1]Extensible Markup Language (XML), version 1.1	Extensible rules for encoding documents	W3C
[2]Namespaces in XML, version 1.1	Rules for uniquely naming elements and attributes	W3C
[3]XML Schema (XSD)[11], version 1.1	Defines the structure, content, and semantics of XML documents	W3C
[4]XML Metadata Interchange (XMI), versions 2.1.1 and 2.1	Standard for exchanging metadata information via XML	OMG
[5]The STEP Application Protocols (AP) ISO 10303	Standard for the Exchange of Product (STEP) model data for systems engineering	ISO

Note Standard Definitions:
[1]XML: Extensible Markup Language (XML) is a simple, very flexible text format derived from Standard Generalized Markup Language (SGML, International Organization for Standardization (ISO) 8879, Information processing – Text and office systems – Standard Generalized Markup Language (SGML)). Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the interoperability of a wide variety of data on the Web and elsewhere.

[2]Namespaces in XML: XML Namespaces[12] are a simple and straightforward way to distinguish names used in XML documents. XML Namespaces provide a simple method for qualifying element and attribute names by associating them with namespaces identified by Uniform Resource Identifier (URI)[13] references.

[3]XML Schema: XML Schemas express shared vocabularies and allow machines to carry out rules made by people. They provide a means for defining the structure, content, and semantics of XML documents in more detail.

[4]OMG9 XML Metadata Interchange (XMI): The main purpose of XMI (ISO/IEC 19503, Information technology – XML Metadata Interchange (XMI)) is to enable easy interoperability of metadata between application development life-cycle tools (such as

[11]XSD is a schema whose purpose is to document the elements of the XML document, their meaning, and their structure.
[12] Namespace in XML1.1, http://www.w3.org/TR/xml-names11/
[13] Uniform Resource Identifier, RFC 3986/STD 66 (2005), http://tools.ietf.org/html/rfc3986

modeling tools based on the Unified Modeling Language (UML) (ISO/IEC 19501, Information technology – Open Distributed Processing – Unified Modeling language (UML)), and metadata repositories/frameworks based on the Meta Object Facility (MOF) (ISO/IEC 19502, Information technology – Meta Object Facility (MOF)), in distributed heterogeneous environments. XMI integrates the following three key industry standards:

(1) XML - Extensible Markup Language.

(2) UML - Unified Modeling Language, an OMG modeling specification (ISO/IEC 19501). (Note: XMI is expressed as UML and is listed here for completeness.)

(3) MOF - Meta Object Facility (ISO/IEC 19502).

[5]The STEP Application Protocols (APs). The STEP APs (e.g., ISO 10303, AP 203, Configuration controlled 3D design; AP 209, Composite and metallic structural analysis and related design; AP 214, Core data for automotive mechanical design processes; AP 233, Systems engineering data representation; or AP 239, Product life cycle support) capture data on components and systems to improve computer-sensible sharing of important product information by supporting the capture, interoperability, and archive systems engineering information across disciplines and organizations.

6.7.3 Contract Language for Data Acquisition

This section provides guidance for ensuring that NASA provides data acquisition requirements to enable the reference and reuse of data through a program's/project's full life cycle. See Appendix A for a list of reference documents.

Data acquisition requirements should be present during RFP/Request for Quote development to establish integrated data architecture for a program's/project's information systems architecture, data architecture, and information systems capability roadmap. The ability to meet data interoperability requirements should be a part of the contract performance metrics on which the contractor has to report and be evaluated throughout contract execution. This recommended contract language should be applied to the Center's data procurement documents.

The following contract language is categorized based on the type of data interoperability being performed. Also included is contract language pertaining to metadata and data transmittal:

a. Proprietary Formatted Data includes data that is generated on a COTS or custom-built application where the application's data format is proprietary to those systems. The contractor should provide or deliver the following:

(1) A definition of the application and version that produced the file.

(2) The native proprietary file itself.

(3) A commonly readable representation of the file (Joint Photographic Experts Group (JPEG, JPG), Graphics Interchange Format (GIF), Portable Network Graphics (PNG), or PDF).

(4) The application (source code and executable), with the concurrence of the NASA Contracting Officer's Technical Representative, required in order to read the native file format for file types where there is no commonly readable representation of the file.

(5) 3D formats, videos, or 3D PDF files.

b. <u>Structured Digital Data from a Source Repository</u> is digital data extraction from a source data repository provided in a computer-readable format that maintains the referential integrity of the data.

(1) The contractor should provide structured, computer-readable data, including all relationships between data, in one of the following formats:

A. Where XML is applied, the contractor should implement an architecture that complies with the applicable NASA IT standards and enable digital data interoperability within NASA domains, W3C XML, W3C Namespaces for XML, and W3C XML Schema.

B. Where XML is applied, the contractor should respond to NASA's requests for IT data items via W3C XML, W3C XML Namespaces, W3C XML Schema, and OMG XMI for Models.

C. For standard XML digital data descriptions to be usable, all producers and consumers of digital data should reference a common data dictionary as defined by NASA. If a NASA standard is not applicable or available, digital data should reference a common data dictionary as defined by the contractor.

D. Database output format (e.g., Structured Query Language (SQL) output).

(2) Structured database output files should include the database type and version (e.g., MySQL[14], version 5.5.1).

(3) The contractor should provide computer-readable definitions of record structure (schema) for both XML and database output formats.

c. <u>Document-Formatted Data</u> describes documents created in established text-processing applications. Document data should be delivered in a digital form in one of the following formats:

---

[14]MySQL is a relational database management system that runs as a server providing multi-user access to a number of databases.

(1) Text-searchable PDF (rather than scanned PDF).

(2) Microsoft® Word and Microsoft® Excel Files.

(3) Other commonly readable format (e.g., a text file (.txt), or rich text format (.rtf)) approved by NASA.

d. Metadata describes the contents of digital files. Metadata should be defined in a computer-readable data format to support automation and facilitate data interoperability. Metadata should be cataloged and published for reuse by all program/project teams to facilitate proper data interoperability.

(1) Acceptable metadata formats include:

A. Comma-Separated Value (CSV) format.

B. Extensible Markup Language (XML) format.

C. Other fully documented, standards-based format approved by NASA such as embedded PDF documents.

(2) Metadata should be delivered with the file to which it is related and reference the filename or other identifying features (e.g., timestamp).

(3) The contractor should apply ISO 10303, AP 233, as a foundation for defining common program/project metadata.

(4) The contractor should apply ISO 10303, AP 239, for common systems engineering metadata.

(5) Metadata used during a program/project phase should be defined more than 90 days prior to its first use.

e. Transmittal of data should be via one of the following options; the data transmission method should be defined in each DRD and RFP statement of work:

(1) View data in contractor systems should support viewing data within the contractor’s electronic system in a format readable by a standard NASA workstation or otherwise approved by NASA. This action will be deemed as a delivery of data in place, and such data will be maintained within the contractor’s system.

(2) Direct entry to NASA system should include direct user entry of data into a NASA-owned system including fields, values, or other information as required by the NASA system.

(3) Upload to NASA file repository should include uploading files, whether document-formatted data or structured data (e.g., SQL output), and completing any associated data as required by the formats (above) and the file repository, including Center, program/project, or other specific repository specified in the DRD.

(4) A system-to-system connection should provide an automated capability to pass computer-readable data between a contractor and NASA system, across contractor and NASA firewalls, via one of several mechanisms, including but not limited to:

A. Representative State Transfer (REST) API for digital data delivery via Internet-based communication (data defined by either a NASA or contractor standard data exchange format).

B. Simple Object Access Protocol (SOAP) API for digital data delivery via Internet-based communication.

C. Fully documented, standards-based API supported by contractor tools approved by NASA.

(5) The contractor should encrypt sensitive information during transmission.

(6) The contractor should utilize the National Institute of Standards and Technology (NIST) 800 Series Publications and FIPS PUB 140-2, Security Requirements for Cryptographic Modules, to meet data encryption requirements.

6.7.4 Contract Language for Defining Data Interoperability

This section provides recommended policy language for guidance in defining data interoperability via industry standards, information systems architecture, and data architecture best practices. A tailored set of the following statements should be included as needed in proposal and planning documents:

a. Data is an essential enabler and should be made visible, accessible, and understandable to any potential user as early as possible in the program's/project's life cycle to support project and mission objectives as constrained by security and access needs.

b. Data assets should be made visible by creating and associating metadata ("tagging"), including but not limited to discovery metadata, for each asset.

c. Developed metadata standards should comply with applicable national and international consensus standards for metadata interoperability. All metadata should be retrievable using the NASA Centers' systems with requirements to access the metadata.

d. Data assets should be accessible by making data available in shared spaces. All data assets should be accessible to all users at the NASA Center except where limited by law, policy, or security classification.

e. Data that is accessible to all users at the NASA Center should conform to that Center's specified data publication methods that are consistent with the Center's current and planned information systems capabilities.

f. To enable trust, data assets should have associated information assurance and security metadata and an identified authoritative contractor and/or Center-level source for the data.

g. Data interoperability should be enabled through business and mission processes reusing the Center's information system capabilities.

h. Semantic and structural agreements for data interoperability should be promoted through communities consisting of data users (producers and consumers) and system developers.

i. Data interoperability concepts and practices should be incorporated into education and awareness training and Center-level processes.

j. Programs/projects should acquire the minimum essential contractor-originated data required to meet all program/project life-cycle requirements.

k. Program/project managers and Technical Authorities should consider data requirements for future procurement needs in a manner that fosters competitive procurement opportunities.

To allow for data interoperability, program/project managers and Technical Authorities should ensure that contractor data delivered to authoritative sources are acquired through the contract and that DRDs specify data to be delivered in specified digital data formats in lieu of "contractor format acceptable" or similar language. Contracts should specify the required delivered data transaction scope, data delivery formats, metadata, naming standards, and data interoperability standards. The definition of data formats and transaction sets should be independent of the method of access or delivery. Regardless of the format of acquired data, the contractor should mark any data provided with less than unlimited rights with the appropriate legend as set forth in FAR 52.227-14, Alternates II and III, Rights in Data—General. Contractors should obtain approval from the NASA Center for all proposed contractual deviations from existing data standards and specifications.

Program/project managers and Technical Authorities should ensure that data is acquired with sufficient access and usage rights to support the full range of program/project needs. This includes, but is not limited to, the need for data to be used by organizations outside NASA, e.g., support contractors and companies bidding on new procurements. In instances where NASA acquires data with less than unlimited rights, the impact to the full program/project is to be assessed. The Contracting Officer and the cognizant legal office should be consulted to develop a strategy to mitigate any adverse impacts associated with the limited or restricted data rights. Program/project managers and Technical Authorities should include language in the statement of work to address the marking of data, e.g., “The contractor should determine the data restriction that applies to each data deliverable and mark or transmit the data restriction in accordance with section X.X of the Data Procurement Document.”

The contractor should incrementally define a:

a. Common vocabulary to support each phase of the program/project life cycle as exit criteria for a program/project life-cycle phase.

b. Common program/project data dictionary to support each phase of the program/project life cycle. This activity should be performed as a program/project phase exit criteria to support the startup of the next phase.

c. Common namespace standards to support each phase of the program/project life cycle as exit criteria for a program/project life-cycle phase.

If a NASA/Center-level common vocabulary, common program/project data dictionary, or common namespace standard exists, then the contract should ensure that developed solutions use these common architecture elements.

7. INTEROPERABILITY AND SUSTAINABILITY

Program/project managers and Technical Authorities are responsible for determining the PDLM interoperability and sustainability requirements and ensuring these are identified across all participating organizations and in all program/project contracts. Inclusion of international partners could have a different set of interoperability and sustainability requirements that would need to be addressed. The following provides guidance to assist with the development of these requirements.

Common problems related to interoperability are listed in table 6, Common Problems Related to Interoperability.

Table 6—Common Problems Related to Interoperability

Issue	Activity/Problem Examples
1. CAD Interoperability	• Determine whether CAD files are ready for use by another engineer in a work group. • Determine level of CAD file usage (see section 6.5.1). • Rework due to incompatible CAD settings.
2. Product Definition	• Identify the software affected by a change in hardware. • Determine whether design data exist and are in a status to be used for procurement*. • Identify all interfaces and navigate to the interface definition document/design object*. • Link part design (e.g., CAD) to one or more specific requirement records**.
3. Part Identification and Naming	• Avoid CAD file name conflicts when combining CAD data from multiple sources. • Compare the definition of two product configurations to be used for analysis. • Reduce error and confusion over nomenclature.
4. Model Definition	• Identify when an alternative representation of a CAD file needs to be updated (e.g., a detail part on a subassembly changes and when other CAD models get updated). • Identify all interfaces and navigate to the interface definition document/design object*.
5. Common Hardware Handling	• Find and replace all instances of a specific part that are in a certain life-cycle state*. • Identify common hardware parts even when provided by different design sources. • Identify parts requiring human flight-rated qualification*.
6. Product Life Cycle	• Confirm the delivery status of product data associated with a contract design deliverable. • Determine whether design data exist and are in a status to be used for procurement*. • Find and replace all instances of a specific part that are in a certain life-cycle state*. • Trace the status changes of a part in life cycle (e.g., changes during "as-manufactured" state)*. • Provide visibility of active engineering changes; show impact on next-higher design objects*. • Identify product objects not in life-cycle state needed to support their higher-level assembly.
7. Product Structure(s)	• Create and approve a product structure(s) from both contractor and NASA sources. • Add the product structure(s) and design data from a subcontractor

Issue	Activity/Problem Examples
	into an existing product structure(s) and control access based on approval/acceptance status. • Update a branch in a product structure(s) (add, delete, revise) based on the product structure(s) received from a contractor. • Identify parts requiring human flight-rated qualification*. • Assess the impact of a part being made obsolete (or otherwise being unavailable). • Identify the detailed configuration of a major test. • Trace the status changes of a part within a given life-cycle state*.

* Relies on more than one standard.
** Standard needed to support IT application integration.

Minimizing customization and adapting external standards can reduce the complexity and would facilitate future upgrades and the exchange of data with external parties. Because process definition is integral to PDLM, the native tool flexibility means that even a no-customization solution can still proliferate complexity and inhibit data sharing.

Consolidation of instances of PDM can reduce the visible number of systems while doing little to achieve the critical goals of improved bi-directional sharing of data internally and externally as stated below:

a. For instance, creation of multiple partitions within a single installation in which each partition owner can reproduce their own processes, data structures, and configuration settings will reduce the number of visible "systems" but not guarantee data exchange.

b. Process terms include the definition of the data objects, the flow steps, the descriptors, user roles, and the decision rules.

c. The processes are implemented by configuring elements into flows, so agreement on the base elements is needed to facilitate data sharing and reduce maintenance costs.

d. The depth of agreement on the base elements and flow configuration is the primary mechanism by which interoperability can be achieved.

In lieu of globally defined and rigid process definitions that have to accommodate all types of work, the core definition of critical processes related to interoperability should be pursued. By looking for commonality around collaboration and data sharing, over-specification is avoided and adaptability is encouraged.

Managing product design and facilitating its use also depends on the design data itself being compliant with a set of practices applied outside the tool. Most notably, the configuration, accuracy, identification, and other settings enacted by engineers within the same tool will affect the ability and effort required to share their data with each other (i.e., as covered in NASA-STD-0007, NASA Computer-Aided Design Interoperability); a perfect PDLM solution cannot overcome incompatibility choices made at the source. The list of processes that need to be discovered and assessed for interoperability begin outside the PDLM tool and extend into manufacturing, procurement, logistics, and mission operations application and systems interfaces.

A program/project manager and the Technical Authority need to address this list of questions that drives process interoperability:

a. How will peer engineers and designers be allowed to share data in support of detailed design?

b. What is sufficient and necessary to define for a specific need (e.g., design, verification, manufacturing):

(1) A part (hardware and software).
(2) An assembly or subsystem.
(3) A system or product.

c. How are parts identified and named to avoid conflict, confusion, and reduce data redundancy?

d. What is the definition of models for a specific purpose (e.g., systems design, part design, envelope, interface checking, thermal and structural analysis, integration, shrink-wrap)?

e. How are the alternate versions of models related and managed, and what is required to ensure traceability and sharing?

f. What practices and resources reduce the effort required to find, use, and integrate approved part definitions (including common hardware)?

g. What state labels, descriptors, and process steps identify the allowed uses, authorized user, and change status of engineering design data during its life cycle (e.g., engineering release, change control, and configuration definition)?

h. What defines a product structure for a given use/audience/life-cycle state, and how is that product structure produced and approved?

7.1 Process Impacts

Three process domains require initial focus for PDLM process interoperability:

a. Data collaboration (the definition and management of data, including models, requirements, CAD parts, CAD model libraries, and other mechanisms that contribute to a complete and valid design through design, development, test, and evaluation (DDT&E)).

b. Engineering release (the very beginning of design data CM where engineering designs are created, checked, and statused).

c. Integrated product structure (the core mechanism around which the relationships between detail product design elements are defined).

Processes are to be implemented concurrently, authorized via institutional mechanisms such as NPRs and standards, and used as expected by trained and supported end users.

Seeking process interoperability also involves inventorying existing mechanisms such as currently deployed capabilities and other application systems (e.g., approved parts list may be in another database), locally approved processes, and the expected practices articulated in Project Plans, contractor practices, or external standards called out in contract language.

Centers have their own documents and implementations inside systems of these processes, and there may be multiples of each. So, interoperability has to be addressed to:

a. Elaborate on the definition of the process domain.

b. Collect data on the current state of things—who has what solutions in place now, in what form, how extensive, what problem does it attack, are they usable outside of any given Center, what is their effectiveness, etc.

c. Create operational concepts (abstract, user-centric interactions) that convey the desired outcome in a solution-independent way.

d. Generate multiple solutions and do trade studies on those solutions leading to recommendation(s) with rationale.

e. Identify the impact of recommendations.

8. PDLM COLLABORATIVE ENVIRONMENT (PCE)

Collaboration should consist of a virtual development environment that allows immediate, secured access to all product data via light-weight visualization tools. Without effective collaboration, team members who are not co-located are forced to share documents using e-mail, zip files, and/or shared server; use multiple tools for collaboration that could manipulate the native format or that could transfer insufficient product information; and always search for the latest version of record.

The PCE creates a development environment that represents all types of “digital” data, including, but not limited to, specifications, requirements, analysis results, and legal entities, and instills a flexible discipline-enabling access to the right information at the right time to the right people. PCE, when leveraged and deployed correctly, can have a major impact on mitigating risk, reducing cycle time, improving quality, and ultimately, cost savings. It also allows for a quick and robust way to make decisions while not incurring any cost on physical prototypes or models. In the end, it is a means toward reducing risk by aiding the decision-making process.

Organizations embracing a PCE vision define the following four themes as what make or break a successful execution:

a. Integrity of the data—having the latest and most correct data from the onset (whether internal or within the contracts that define the exchange of information). The data has to be complete as well as consistent, making sure that the right information is available. DRDs will define the level of depth and detail needed, as well as how that information can be exchanged to ensure integrity is enforced.

b. Virtual access anytime/anywhere—global sharing, including security and safeguarding of data, as well as supply chain enablement, is critical to handling complex systems from both a push and pull perspective as part of operational workflow. This capability is critical for operations such as engineering release or change management, making sure the data is available at the right time to the right people. The completeness and integrity of the data and ability to access information 24 hours, 7 days a week are expected.

c. Discipline in integrated product definition-driven development—stemming from the integrated product structure (including options and variants). Once the structure is defined, management of configurations and changes need to be scalable as well as enforced.

d. Visualization—just beginning to be referred to as "virtualization"—using light-weight and intelligent visualization software that is capable of total product representation and animation for form and fit decisions, analysis, management, and assembly.

Successful product development for NASA relies on some key capabilities as follows:

a. Visibility and management of requirements throughout the product development process.

b. Access to a more complete "data model" at any time during the development process that represents an intelligent and fully defined artifact versus an amassing of multiple documents from remote resources.

c. More informed decision making for the appropriate selection of design strategy.

d. Ability to better leverage and enable the different program/project levels and primes by bringing them closer to the product development life-cycle phase and decision process.

e. Ability to reuse products and processes and incorporate new requirements as needed/when needed.

f. Flexibility for domain autonomy built into product design and manufacturing processes.

g. Validation of functional product and process capability to ensure that products meet the needs of the Agency and that processes work effectively.

While PDLM can provide value and help achieve some of these technical goals, it will not be totally successful without changes in many domains. Development of a PCE is essential in fully achieving these goals.

As a part of executing a PCE strategy, desired data end states have to be converted into more specific objectives as follows:

a. Pre-release Data Sharing—When serving as systems integrator, engineering data have to be shared with other engineering groups, including to and from primes, prior to formal engineering release. Vertical data movement (e.g., downward for requirements, upward for analysis results) supports decision making but does not support the horizontal and diagonal flow of data, especially CAD and other specialized data. The question was how to keep track of which version of the model that was used and what it was used for. Today, if a problem is found in that model, the capabilities to share such data may outstrip the organizational capacity to facilitate and monitor it. MBSE, along with M&S, does a very good job of defining the infrastructure that defines access to and what is required to answer many of these questions; but it is the PCE framework (as defined by the other initiatives) that makes the access and sharing possible.

b. Distributed, Non-Document-Centric Authority—Product structures and designs are just some of the data that are managed in systems. They cannot serve their purpose if the electronic version is not authoritative. On many projects, authority for design is distributed unequally: the number of data objects to be reviewed and approved increases non-linearly as one goes DOWN the product hierarchy. Furthermore, the distributed vehicle elements are simultaneously in different design states; highly distributed authority combined with the interlinked data structures of CAD complicates the dynamics of intra-element flow of design data.

c. Process, Tool, and Use Synchronization—In PCE, the processes and workflows are implemented in the tool suite and provide real value when they are actually used. The engineering release definition, as well as NPR 7123.1, defines this model; and PDLM can act as the framework to ensure that the right data is available at the right time.

d. Handling Data Objects—The larger picture of significant design data being created by primes and partners really shines a spotlight on scalability and usability challenges experienced today around handling objects within PDLM environments. Definition of the need for the model and its attributes is blurred; therefore, native data (which can be very large in itself) carries a lot of unnecessary baggage that will never be used by the recipient. New ways of thinking are needed about what it means to make a "delivery," to "receive" and/or "use" a model (or representation) which then needs to be properly reflected in the DRD clause, to "release" a design, and how to deal with ownership, change and configuration control around complex data entities that are still actively being defined and do not behave like PDF files or drawings.

e. Having a Voice—NASA has dynamic needs because the mission is research, science, and exploration. The mission is to do things that have not been done before. When processes are linked to tools and one cannot do the job without the data that is in them, a move toward processes enabled by technology is necessary. Who gets to decide on the requirements for the application? Who gets to influence the decision on the trade analysis results? Who gets to move their problem to the top of the "to do" list? Governance has to recognize the needs of different users to be heard at different times, and the infrastructure has to support it.

f. Data Usability and Collaboration—Capability deployment that maps to defined user communities. Achieving this landscape will require both technical and process solutions to be designed, built, configured, tested, deployed, used, and supported. Additional attention to enable the move from document centricity to data will require focus on new practices, including contract language (DRD clauses) and supporting tools, around specification of technical data packages and the conduct of data and CM.

9. PDLM PLAN DEVELOPMENT

PDLM is a critical element with technical and programmatic implications. A program/project manager and the Technical Authority have the responsibility to determine the following:

a. How much should be risked (or how much should the program/project pay) to ensure the relevant data exist and are accessible, discoverable, and understandable to support such events as an in-flight anomaly 10 or more years in the future?

b. Where should the Agency and the program/project invest their limited attention and resources on data during the development cycle?

The PDLM Plan documents the foundation for these decisions. The program/project managers and the Technical Authority sign and release the PDLM Plan prior to the completion of the System Definition Review (SDR). A program's/project's compliance with NPR 7120.9 is verified by submission of the initial preliminary PDLM Plan. A program/project manager and the Technical Authority require support in the development of the PDLM strategy and the PDLM Plan. The following are support role definitions that may assist in the development of a program/project management PDLM team:

a. PDLM Solution Architect/PDLM Program/Project Manager and the Technical Authority

(1) Technical program/project strategy and oversight leadership support and advising the program/project manager and the Technical Authority on how to facilitate knowledge transfer, education, technical direction, and consultation from program/project beginning to end of life.

(2) Long term—focus on the end-to-end enterprise level system/solution design, and roll-out strategies that drive toward implementation delivery (associated with) and sustainment of the following:

A. Identification and development of the technology roadmap based on desired end state.

B. Process alignment and optimization by proper application of enabling technologies.

C. Design consultation including security, authentication, authorization, and integration.

D. Product capability deployment, features, and functionality.

E. Automation processes and methodologies.

F. Knowledge management, best practices, and reuse strategies.

G. Perform the role of the senior technical designer on the project.

H. Drive the linkage/enablement of the designed architecture to domain processes, such as CM, engineering release, change management, etc.

I. Lead the development of the system and application architecture definition and application configuration including:

i. Hardware and software components of the system.
ii. Interfaces and interoperability.
iii. Infrastructure and security.
iv. Performance and reliability.
v. Administration and access control.
vi. Maintenance and sustainment over time.

(3) Near term**—**focus on architecture implementation of next delivery milestones by the following:

A. Lead and drive in the development and review of all related infrastructure, architecture, and design components.

B. Lead the development of the system description and architecture design.

C. Lead the development of any customization designs.

D. Lead in the technical review of the use-case scenarios, requirements, and deployment decisions.

E. Chair and conduct design, code, and configuration reviews allowing project management to enforce consistency with the overall technical architecture.

b. PDLM Tool Implementation Lead

(1) Conduct the installation of PDLM tool (software, etc.) and third party applications.

(2) Configure and sustain the tools/applications following installation of the tool by the Agency CIO, or designee, including:

A. Access control.
B. Data/folder structures.
C. Workflow design and configuration.
D. System test and validation.
E. Data migration.
F. Data retrieval from external systems.
G. Development of test scripts.

c. PDLM Process Lead

(1) Lead the definition of overall business requirements (key process areas and indicators).

(2) Lead in the gathering and analysis of solution and system capability/functionality.

(3) Lead in the assessment of as-is business process documentation and gap identification of process documentation.

(4) Lead in the definition and documentation of business use cases via concepts of operations.

(5) Interface with the solution architects regarding system capability to ensure defined processes are achievable within the technology and within program/project milestone constraints.

(6) Analyze complex customer business process requirements to evaluate alternatives and identify options within the solution system architecture.

(7) Ensure the development of process definition and use-case documentation.

Table 7, PDLM Plan Section Development, lists the PDLM sections, identifies the PDLM contributors, and lists the agreements that should be reached for each PDLM area.

9.1 PDLM Plan Schedule

The content of the PDLM Plan is affected by both the program's/project's phasing and by current and planned changes to the IT used to implement PDLM (e.g., updates and capabilities committed to by PDLM tool providers). The Program Plan needs to be updated annually or at a major Key Decision Point (KDP). In the case of product life cycle, the plan needs to be updated through KDP F (close out/archival of data). At KDP C, the programs/projects are beginning to understand the data roadmap (Architecture) and inter-dependencies between the suppliers and NASA. The following table 8, PDLM Plan Development Guidelines, is a recommended guideline for the development of the PDLM framework and element information that is to be captured in the PDLM Plan.

Table 7—PDLM Plan Section Development

Section No.	PDLM Plan Section	Responsibility/Competency	Action
1.0	PROGRAM/PROJECT OVERVIEW	Program/Project Management; Center(s) Management; Center(s) CIO; Center(s) Engineering Department	Contributors agree on the stated performance objectives for the Program/Project PDLM.
1.1	Introduction		
1.2	Objectives		
1.3	Solution Summary		
1.4	Assumptions, Limitations, and Constraints		
1.5	Responsible Organizations, Governance, and Plan Update Timing		
2.0	PROGRAM/PROJECT PDLM REQUIREMENTS TRACEABILITY	PDLM Program/Project Manager and Technical Authority	Requirement Traceability Matrix identifies compliance that the PDLM System meets the requirements of the NPR.
2.1	Requirements Traceability Matrix		
2.2	“Do-Not-Apply” Justifications		
2.3	Future Work Discussion		
3.0	PDLM STRATEGY	PDLM Program/Project Manager and Technical Authority	Work with engineering and stakeholders to define PDLM needs. Determine data exchange between disparate PDLMs (if needed) and determine if gap analysis to resolve issues is needed.
3.1	PDLM Strategy		
3.2	PDM Approach		
3.3	PLM Approach		
4.0	PDLM IMPLEMENTATION DETAILS		

Section No.	PDLM Plan Section	Responsibility/Competency	Action
4.1	Processes	PDLM Process Lead	Program/project, and Center organizations review, modify, and determine if gap analysis is needed for existing Center processes for use by the program/project.
4.2	User Communities	PDLM Program/Project Manager and Technical Authority	Determine the users and their locations of the PDLM system(s).
4.3	PDLM Tools	PDLM Tool Implementation Lead	Works with Center CIO to identify existing PDLM system/tools and with program/project to determine if current tools will satisfy program/project needs.
4.4	Process Mapping	PDLM Process Lead	Identifies functional organizational processes which are to be used by the program/project and determines if gap analysis is needed. Also, identifies the interfaces between related processes (e.g., program/project CM to Center CM).
4.5	Data Architecture	PDLM Program/Project Manager and Technical Authority	The NASA Chief Engineer provides the program/project with its PDLM architecture document(s) to assure program/project PDLM needs will be met. PDLM program/project manager and Technical Authority identify data relationships between different object types.
4.6	Product Breakdown Structure (Bill of Material) Process	PDLM Process Lead	Determines program/project product breakdown structure needs; determines if current architecture will meet needs.
4.7	Specialized Data Type Process	PDLM Process Lead	Identifies the PDLM data objects and associated attributes needed to support the programs/projects. Determines what the data interoperability requirements are.

Section No.	PDLM Plan Section	Responsibility/Competency	Action
4.8	Specialized Libraries and Part Data	PDLM Process Lead	Determines what types of libraries and part objects are required to support the program/project. Defines the part and document identification schema.
4.9	Engineering Release, Configuration Management, and Change Control Processes	PDLM Process Lead	Works with CM organization to define change control process. Defines, and documents as part of the PDLM Plan, what product data is released, when that product data is to be released, the events that necessitate product data change, and the processes that provide the visibility of the product data configuration life cycle, for internally and externally produced product data.
4.10	Data Management Process	PDLM Process Lead	Identifies the PDLM data objects and associated attributes needed to support the programs/projects.
4.11	Information Security	PDLM Program/Project Manager and Technical Authority	Work with Center CIO to identify applicable security plans for applications and systems to be used.
D.4	APPENDICES		

Table 8—PDLM Plan Development Guidelines

Preliminary PDLM Plan Signed and Released

		Formulation				Implementation		
PDLM Elements	PDLM Plan Sections per NPR 7120.9	Pre-Phase A Concept Studies	Phase A Concept & Technology Development	Phase B Prelim Design & Technology Completion	Phase C Final Design & Fabrication	Phase D Sys Assembly, Int &Test, Launch & Checkout	Phase E Operation & Sustainment	Phase F Closeout
		KDP 0	KDP I	KDP II	KDP III	KDP IV	KDP V	KDP VI
Security Architecture	1, 3.1, 4,2	Preliminary	Preliminary	Baseline	Update	Update	Update	
Information Support System Architecture	1, 3.1, 4.2, 4.11	Preliminary	Preliminary	Baseline	Update	Update	Update	
Data Architecture	2, 3.1, 4.2, 4.3, 4.5	Preliminary	Preliminary	Baseline	Update	Update	Update	
Process Architecture	2, 3.1, 4.1, 4.4	Preliminary	Preliminary	Baseline	Update	Update	Update	
Requirements Management	2.1	Preliminary	Baseline	Update	Update			
Configuration Management	4.9	*	Baseline	Update	Update	Update	Update	Update
Risk Management	3.3	*	Preliminary	Baseline	Update	Update	Update	

		Formulation				Implementation		
PDLM Elements	**PDLM Plan Sections per NPR 7120.9**	**Pre-Phase A** Concept Studies	**Phase A** Concept & Technology Development	**Phase B** Prelim Design & Technology Completion	**Phase C** Final Design & Fabrication	**Phase D** Sys Assembly, Int &Test, Launch & Checkout	**Phase E** Operation & Sustainment	**Phase F** Closeout
		KDP 0	**KDP I**	**KDP II**	**KDP III**	**KDP IV**	**KDP V**	**KDP VI**
Product Data Management	**3.2, 4.10**	*	**Preliminary**	**Baseline**	**Update**	**Update**	**Update**	**Update**
Parts Management	**4.7, 4.8**	*	*	**Preliminary**	**Baseline**	**Update**	**Update**	**Update**
CAD Data Management	**4.7, 4.8**	*	**Preliminary**	**Baseline**	**Update**	**Update**	**Update**	
Product Structure (Bill of Material)	**4.6**	*	**Preliminary**	**Preliminary**	**Baseline**	**Update**	**Update**	
Models-Based Design, including Models and Simulations	**4.7**	**Preliminary**	**Preliminary**	**Baseline**	**Update**	**Update**	**Update**	
Digital Data Standards and Contract Language	**4.7, 4.9**	**Preliminary**	**Preliminary**	**Baseline**	**Update**	**Update**		
Interoperability and Sustainability	**4.5, 4.7, 4.9**	**Preliminary**	**Preliminary**	**Baseline**	**Update**	**Update**	**Update**	

*These items are to be addressed in the PDLM Plan but are expected to be an in-work or to-be-determined state with planned resolution.

APPENDIX A

REFERENCES

A.1 Purpose and/or Scope

The purpose of this appendix is to provide additional guidance available in the references below.

A.2 References

The following references are recommended for further guidance[15]:

a. MBSE

(1) NASA Models-Based Systems Engineering Group, https://nen.nasa.gov/web/se/mbse.

(2) INCOSE (International Council on Systems Engineering), www.incose.org.

(3) INCOSE (2007) MBSE Initiative Presentation, http://www.incose.org/enchantment/docs/07docs/07jul_4mbseroadmap.pdf.

(4) International Council on Systems Engineering (INCOSE), *Systems Engineering Vision 2020*, Version 2.03, TP-2004-004-02, September 2007.

(5) Requirements Exchange: ReqIF Requirement Interchange Format- Object Management Group™ standard http://www.omg.org/spec/ReqIF/.

b. CAD Data Management

PROSTEP AG, White Paper, "3D Formats in the Field of Engineering – A Comparison, http://www.us.pdfgenerator3d.com/nc/en/product/white-paper.html.

c. Data Interoperability

ISO 8879	Information processing – Text and office systems – Standard Generalized Markup Language (SGML)
ISO 16792	Technical product documentation—Digital product definition data practices

[15] If Web links are not accessible from this document, copy and paste the Web address into the address block after accessing the Internet.

ISO/IEC 19501	Information technology – Open Distributed Processing – Unified Modeling Language (UML)
ISO/IEC 19502	Information technology – Meta Object Facility (MOF)
ISO/IEC 19503	Information technology – XML Metadata Interchange (XMI)
ISO/IEC 24765: 2009	Systems and software engineering vocabulary

d. Contract Language

ISO 10303, AP 203	Standard for the Exchange of Product (STEP) Model Data, AP 203, Configuration controlled 3D design
ISO 10303, AP 209	Standard for the Exchange of Product (STEP) Model Data, AP 209, Composite and metallic structural analysis and related design
ISO 10303, AP 214	Standard for the Exchange of Product (STEP) Model Data, AP 214, Core data for automotive mechanical design processes
ISO 10303, AP 233	Standard for the Exchange of Product (STEP) Model Data, AP 233, Systems engineering data representation
ISO 10303, AP 239	Standard for the Exchange of Product (STEP) Model Data, AP 239, Product life-cycle support
	NIST 800 Series Publications

e. Contract Language for Defining Data Interoperability

FAR 52.227-14, Alternates II and III	Rights in Data—General

f. Exchanging and Distributing 3D Models

ISO 10303	STEP (Standard for the Exchange of Product model data) - ISO global standard, http://pdesinc.aticorp.org/step_info_links.html, http://step.nasa.gov/
ISO/DIS 14306	Industrial automation systems and integration -- JT file format specification for 3D visualization, www.iso.org

ISO 32000-1:2008	3D PDF (3D Portable Document Format) – Adobe Acrobat, www.iso.org 3D XML (3D eXtensible Markup Language) - Dassult Systemes, http://www.3ds.com/products/3dvia/3d-xml/overview

g. Other Neutral Formats not listed in Table 3

ANS US PRO/ IPO-100-1996	IGES (Initial Graphics Exchange Specification), Version 5.3, United States Product Data Association (US PRO). Note: No further work is being done on this standard, and US PRO closed in 2006.
ISO/IEC 8632: 1999	CGM (Computer Graphics Metafile), www.iso.org
ISO/IEC 19776-2:2011	COLLADA (COLLAborative Design Activity) – Khronous Group, collada.org, X3D (eXtensible 3D), www.iso.org
	DXF (Drawing Exchange Format) – AutoDesk, usa.autodesk.com

h. Institute of Electrical and Electronics Engineers (IEEE)

IEEE-STD-610.12-1990	IEEE Standard Glossary of Software Engineering Terminology

i. Federal

FIPS PUB 140-2	Security Requirements for Cryptographic Modules

j. Department of Defense (DoD)

DoD Directive 8320.03	Unique Identification (UID) Standards for a Net-Centric Department of Defense

k. NASA

NPD 1000.0	NASA Governance and Strategic Management Handbook
NPD 7120.4	NASA Engineering and Program/Project Management Policy
NPR 2830.1	NASA Enterprise Architecture Procedures
NPR 7120.5	NASA Space Flight Program and Project Management Requirements

NASA-STD-0005 NASA Configuration Management (CM) Standard

NASA/SP-6107 Human Exploration of Mars: The Reference Mission of the Mars Exploration Study Team

MSFC-STD-3528 Computer-Aided Design (CAD) Standard

NIMA-RPT-002 Data Integrity in NASA Programs and Projects (White Paper), https://nen.nasa.gov

NIMA-RPT-004 Future Data Exchange for NASA Programs (White Paper), authors Patrick L. Anderson, Timothy A. Deibel, and Kevin R. Long, developed by the Constellation Information Systems Office, https://nen.nasa.gov